INSTITUTE OF AERONAUTICAL ENGINEERING

(Autonomous)

(Approved by AICTE | NAAC Accreditation with 'A' Grade | Accredited by NBA | Affiliated to JNTUH)

Dundigal, Hyderabad - 500 043, Telangana

OUTCOME BASED EDUCATION
WITH
CHOICE BASED CREDIT SYSTEM

BACHELOR OF TECHNOLOGY
ELECTRICAL AND ELECTRONICS ENGINEERING

ACADEMIC REGULATIONS, COURSE STRUCTURE AND SYLLABI
UG20

B.Tech Regular Four Year Degree Program
(for the batches admitted from the academic year 2020 - 2021)
&

B.Tech (Lateral Entry Scheme)
(for the batches admitted from the academic year 2021 - 2022)

These rules and regulations may be altered/changed from time to time by the academic council
FAILURE TO READ AND UNDERSTAND THE RULES IS NOT AN EXCUSE

VISION

To bring forth professionally competent and socially sensitive engineers, capable of working across cultures meeting the global standards ethically.

MISSION

To provide students with an extensive and exceptional education that prepares them to excel in their profession, guided by dynamic intellectual community and be able to face the technically complex world with creative leadership qualities.

Further, be instrumental in emanating new knowledge through innovative research that emboldens entrepreneurship and economic development for the benefit of wide spread community.

QUALITY POLICY

Our policy is to nurture and build diligent and dedicated community of engineers providing a professional and unprejudiced environment, thus justifying the purpose of teaching and satisfying the stake holders.

A team of well qualified and experienced professionals ensure quality education with its practical application in all areas of the Institute.

PROGRAM OUTCOMES (PO's)

Engineering Graduates will be able to:

PO1: **Engineering Knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2: **Problem Analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3: **Design/Development of Solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO4: **Conduct Investigations of Complex Problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5: **Modern Tool Usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO6: **The Engineer and Society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7: **Environment and Sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8: **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO9: **Individual and Team Work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO10: **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11: **Project Management and Finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one’s own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO12: **Life-Long Learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

CONTENTS

"Take up one idea.
Make that one idea your life-think of it, dream of it, live on that idea.
Let the brain muscles, nerves, every part of your body be full of that idea and just leave every other idea alone. This is the way to success"

Swami Vivekananda

PRELIMINARY DEFINITIONS AND NOMENCLATURES

AICTE: Means All India Council for Technical Education, New Delhi.

Autonomous Institute: Means an institute designated as Autonomous by University Grants Commission (UGC), New Delhi in concurrence with affiliating University (Jawaharlal Nehru Technological University, Hyderabad) and State Government.

Academic Autonomy: Means freedom to an institute in all aspects of conducting its academic programs, granted by UGC for Promoting Excellence.

Academic Council: The Academic Council is the highest academic body of the institute and is responsible for the maintenance of standards of instruction, education and examination within the institute. Academic Council is an authority as per UGC regulations and it has the right to take decisions on all academic matters including academic research.

Academic Year: It is the period necessary to complete an actual course of study within a year. It comprises two main semesters i.e., (one odd + one even) and one supplementary semester.

Branch: Means specialization in a program like B.Tech degree program in Aeronautical Engineering, B.Tech degree program in Computer Science and Engineering etc.

Board of Studies (BOS): BOS is an authority as defined in UGC regulations, constituted by Head of the Organization for each of the departments separately. They are responsible for curriculum design and updation in respect of all the programs offered by a department.

Backlog Course: A course is considered to be a backlog course, if the student has obtained a failure grade (F) in that course.

Basic Sciences: The courses offered in the areas of Mathematics, Physics, Chemistry etc., are considered to be foundational in nature.

Betterment: Betterment is a way that contributes towards improvement of the students' grade in any course(s). It can be done by either (a) re-appearing or (b) re-registering for the course.

Commission: Means University Grants Commission (UGC), New Delhi.

Choice Based Credit System: The credit based semester system is one which provides flexibility in designing curriculum and assigning credits based on the course content and hours of teaching along with provision of choice for the student in the course selection.

Certificate Course: It is a course that makes a student to have hands-on expertise and skills required for holistic development in a specific area/field.

Compulsory course: Course required to be undertaken for the award of the degree as per the program.

Continuous Internal Examination: It is an examination conducted towards sessional assessment.

Core: The courses that are essential constituents of each engineering discipline are categorized as professional core courses for that discipline.

Course: A course is a subject offered by a department for learning in a particular semester.

Course Outcomes: The essential skills that need to be acquired by every student through a course.

Credit: A credit is a unit that gives weight to the value, level or time requirements of an academic course. The number of 'Contact Hours' in a week of a particular course determines its credit value. One credit is equivalent to one lecture/tutorial hour per week.

Credit point: It is the product of grade point and number of credits for a course.

Cumulative Grade Point Average (CGPA): It is a measure of cumulative performance of a student over all the completed semesters. The CGPA is the ratio of total credit points secured by a student in various courses in all semesters and the sum of the total credits of all courses in all the semesters. It is expressed up to two decimal places.

Curriculum: Curriculum incorporates the planned interaction of students with instructional content, materials, resources, and processes for evaluating the attainment of Program Educational Objectives.

Department: An academic entity that conducts relevant curricular and co-curricular activities, involving both teaching and non-teaching staff, and other resources in the process of study for a degree.

Detention in a Course: Student who does not obtain minimum prescribed attendance in a course shall be detained in that particular course.

Dropping from Semester: Student who doesn't want to register for any semester can apply in writing in prescribed format before the commencement of that semester.

Elective Course: A course that can be chosen from a set of courses. An elective can be Professional Elective and / or Open Elective.

Evaluation: Evaluation is the process of judging the academic performance of the student in her/his courses. It is done through a combination of continuous internal assessment and semester end examinations.

Experiential Engineering Education (ExEEd): Engineering entrepreneurship requires strong technical skills in engineering design and computation with key business skills from marketing to business model generation. Our students require sufficient skills to innovate in existing companies or create their own.

Grade: It is an index of the performance of the students in a said course. Grades are indicated by alphabets.

Grade Point: It is a numerical weight allotted to each letter grade on a 10 - point scale.

Honours: An Honours degree typically refers to a higher level of academic achievement at an undergraduate level.

Institute: Means Institute of Aeronautical Engineering, Hyderabad unless indicated otherwise by the context.

Massive Open Online Courses (MOOC): MOOC courses inculcate the habit of self learning. MOOC courses would be additional choices in all the elective group courses.

Minor: Minor are coherent sequences of courses which may be taken in addition to the courses required for the B.Tech degree.

Pre-requisite: A specific course or subject, the knowledge of which is required to complete before student register another course at the next grade level.

Professional Elective: It indicates a course that is discipline centric. An appropriate choice of minimum number of such electives as specified in the program will lead to a degree with specialization.

Program: Means, UG degree program: Bachelor of Technology (B.Tech); PG degree program: Master of Technology (M.Tech) / Master of Business Administration (MBA).

Program Educational Objectives: The broad career, professional and personal goals that every student will achieve through a strategic and sequential action plan.

Project work: It is a design or research based work to be taken up by a student during his/her final year to achieve a particular aim. It is a credit based course and is to be planned carefully by the student.

Re-Appearing: A student can reappear only in the semester end examination for theory component of a course, subject to the regulations contained herein.

Registration: Process of enrolling into a set of courses in a semester of a program.

Regulations: The regulations, common to all B.Tech programs offered by Institute, are designated as "IARE Regulations – R20" and are binding on all the stakeholders.

Semester: It is a period of study consisting of 15 to 18 weeks of academic work equivalent to normally 90 working days. Odd semester commences usually in July and even semester in December of every year.

Semester End Examinations: It is an examination conducted for all courses offered in a semester at the end of the semester.

S/he: Means "she" and "he" both.

Student Outcomes: The essential skill sets that need to be acquired by every student during her/his program of study. These skill sets are in the areas of employability, entrepreneurial, social and behavioral.

University: Means Jawaharlal Nehru Technological University Hyderabad (JNTUH), Hyderabad, is an affiliating University.

Withdraw from a Course: Withdrawing from a course means that a student can drop from a course within the first two weeks of odd or even semester (deadlines are different for summer sessions). However, s/he can choose a substitute course in place of it, by exercising the option within 5 working days from the date of withdrawal.

FOREWORD

The autonomy is conferred to Institute of Aeronautical Engineering (IARE), Hyderabad by University Grants Commission (UGC), New Delhi based on its performance as well as future commitment and competency to impart quality education. It is a mark of its ability to function independently in accordance with the set norms of the monitoring bodies including J N T University Hyderabad (JNTUH), Hyderabad and AICTE, New Delhi. It reflects the confidence of the affiliating University in the autonomous institution to uphold and maintain standards it expects to deliver on its own behalf. Thus, an autonomous institution is given the freedom to have its own **curriculum, examination system** and **monitoring mechanism**, independent of the affiliating University but under its observance.

IARE is proud to win the credence of all the above bodies monitoring the quality in education and has gladly accepted the responsibility of sustaining, if not improving upon the standards and ethics for which it has been striving for more than a decade in reaching its present standing in the arena of contemporary technical education. As a follow up, statutory bodies such as Academic Council and Board of Studies (BOS) are constituted with the guidance of the Governing Body of the institute and recommendations of the JNTUH to frame the regulations, course structure, and syllabi under autonomous status.

The autonomous regulations, course structure, and syllabi have been prepared after prolonged and detailed interaction with several expertise solicited from academics, industry and research, in accordance with the vision and mission of the institute in order to produce a quality engineering graduate to the society.

All the faculty, parents, and students are requested to go through all the rules and regulations carefully. Any clarifications needed are to be sought at appropriate time and from the principal of the institute, without presumptions, to avoid unwanted subsequent inconveniences and embarrassments. The cooperation of all the stake holders is requested for the successful implementation of the autonomous system in the larger interests of the institute and brighter prospects of engineering graduates.

PRINCIPAL

INSTITUTE OF AERONAUTICAL ENGINEERING

(Autonomous)

ACADEMIC REGULATIONS – UG.20

B.Tech. Regular Four Year Degree Program
(for the batches admitted from the academic year 2020 - 2021)
&
B.Tech. (Lateral Entry Scheme)
(for the batches admitted from the academic year 2021 - 2022)

For pursuing four year undergraduate Bachelor of Technology (B.Tech) degree program of study in engineering offered by Institute of Aeronautical Engineering under Autonomous status.

A student shall undergo the prescribed courses as given in the program curriculum to obtain his/her degree in major in which he/she is admitted with 160 credits in the entire program of 4 years. Additional 20 credits can be acquired for the degree of B.Tech with **Honours or additional Minor in Engineering**. These additional 20 credits will have to be acquired with massive open online courses (MOOCs), to tap the zeal and excitement of learning beyond the classrooms. This creates an excellent opportunity for students to acquire the necessary skill set for employability through massive open online courses where the rare expertise of world famous experts from academics and industry are available.

Separate certificate will be issued in addition to major degree program mentioning that the student has cleared Honours / Minor specialization in respective courses.

1. CHOICE BASED CREDIT SYSTEM

The credit based semester system provides flexibility in designing program curriculum and assigning credits based on the course content and hours of teaching. The Choice Based Credit System (CBCS) provides a 'cafeteria' type approach in which the students can take courses of their choice, learn at their own pace, undergo additional courses and acquire more than the required credits, and adopt an interdisciplinary approach to learning.

A course defines learning objectives and learning outcomes and comprises lectures / tutorials / laboratory work / field work / project work / comprehensive examination / seminars / assignments / MOOCs / alternative assessment tools / presentations / self-study etc., or a combination of some of these. Under the CBCS, the requirement for awarding a degree is prescribed in terms of number of credits to be completed by the students.

2. MEDIUM OF INSTRUCTION

The medium of instruction shall be **English** for all courses, examinations, seminar presentations and project work. The program curriculum will comprise courses of study as given in course structure, in accordance with the prescribed syllabi.

3. PROGRAMS OFFERED

Presently, the institute is offering Bachelor of Technology (B.Tech) degree programs in eleven disciplines. The various programs and their two-letter unique codes are given in Table 1.

Table 1: B.Tech Programs offered

S. No	Name of the Program	Title	Code
1	Aeronautical Engineering	AE	07
2	Computer Science and Engineering	CS	05
3	Computer Science and Engineering (AI & ML)	CA	34
4	Computer Science and Engineering (Data Science)	CD	35
5	Computer Science and Engineering (Cyber Security)	CC	36
6	Computer Science and Information Technology	CI	37
7	Information Technology	IT	06
8	Electronics and Communication Engineering	EC	04
9	Electrical and Electronics Engineering	EE	02
10	Mechanical Engineering	ME	03
11	Civil Engineering	CE	01

4. SEMESTER STRUCTURE

Each academic year is divided into three semesters, TWO being **MAIN SEMESTERS** (one odd + one even) and ONE being a **SUPPLEMENTARY SEMESTER**. Main semesters are for regular class work. Supplementary Semester is primarily for failed students i.e. registration for a course for the first time is generally not permitted in the supplementary semester.

4.1 Each main semester shall be of 21 weeks (Table 1) duration and this period includes time for registration of courses, course work, examination preparation, and conduct of examinations.

4.2 Each main semester shall have a minimum of 90 working days.

4.3 The supplementary semester shall be a fast track semester consisting of eight weeks and this period includes time for registration of courses, course work, and examination preparation, conduct of examinations, assessment, and declaration of final results.

4.4 All subjects may not be offered in the supplementary semester. The student has to pay a stipulated fee prescribed by the institute to register for a course in the supplementary semester. The supplementary semester is provided to help the student in not losing an academic year. It is optional for a student to make use of supplementary semester. **Supplementary semester is a special semester and the student cannot demand it as a matter of right** and will be offered based on availability of faculty and other institute resources.

4.5 The institute may use **supplementary semester** to arrange add-on courses for regular students and / or for deputing them for practical training / FSI model. A student can register for a maximum number of 15 credits during a supplementary semester.

The registration for the supplementary semester (during May – July, every year) provides an opportunity to students to clear their backlogs ('F' grade) or who are prevented from appearing for SEE examinations due to shortage of attendance less than 65% in each course ('SA' Grade) in the earlier semesters or the courses which he / she could not register (Drop / Withdraw) due to any reason.

Students will not be permitted to register for more than 15 credits (both I and II semester) in the supplementary semester. Students required to register for supplementary semester courses are to pay a nominal fee within the stipulated time. A separate circular shall be issued at the time of supplementary semester.

It will be optional for a student to get registered in the course(s) of supplementary semester; otherwise, he / she can opt to appear directly in supplementary examination. However, if a student gets registered in a course of supplementary semester, then it will be compulsory for a student to fulfill attendance criterion (≥90%) of supplementary semester and he / she will lose option to appear in immediate supplementary examination.

The students who have earlier taken SEE examination and register afresh for the supplementary semester may revoke the CIA marks secured by them in their regular/earlier attempts in the same course. Once revoked, the students shall not seek restoration of the CIA marks.

Supplementary semester will be at an accelerated pace e.g. one credit of a course shall require two hours/week so that the total number of contact hours can be maintained same as in normal semester.

Instructions and guidelines for the supplementary semester course:

- A minimum of 36 to 40 hours will be taught by the faculty for every course.
- Only the students registered and having sufficient percentage of attendance for the course will be permitted to write the examination.
- The assessment procedure in a supplementary semester course will be similar to the procedure for a regular semester course.
- Student shall register for the supplementary semester as per the schedule given in academic calendar.
- Once registered, students will not be allowed to withdraw from supplementary semester.

4.6 The academic calendar shown in Table 2 is declared at the beginning of the academic year.

Table 2: Academic Calendar

<table>
<tr><td rowspan="6">FIRST SEMESTER (21 weeks)</td><td>I Spell Instruction Period</td><td>8 weeks</td><td rowspan="5">19 weeks</td></tr>
<tr><td>I Continuous Internal Assessment Examinations (Mid-term)</td><td>1 week</td></tr>
<tr><td>II Spell Instruction Period</td><td>8 weeks</td></tr>
<tr><td>II Continuous Internal Assessment Examinations (Mid-term)</td><td>1 week</td></tr>
<tr><td>Preparation and Practical Examinations</td><td>1 week</td></tr>
<tr><td colspan="2">Semester End Examinations</td><td>2 weeks</td></tr>
<tr><td colspan="3">Semester Break and Supplementary Exams</td><td>2 weeks</td></tr>
<tr><td rowspan="6">SECOND SEMESTER (21 weeks)</td><td>I Spell Instruction Period</td><td>8 weeks</td><td rowspan="5">19 weeks</td></tr>
<tr><td>I Continuous Internal Assessment Examinations (Mid-term)</td><td>1 week</td></tr>
<tr><td>II Spell Instruction Period</td><td>8 weeks</td></tr>
<tr><td>II Continuous Internal Assessment Examinations (Mid-term)</td><td>1 week</td></tr>
<tr><td>Preparation & Practical Examinations</td><td>1 week</td></tr>
<tr><td colspan="2">Semester End Examinations</td><td>2 weeks</td></tr>
<tr><td colspan="3">Summer Vacation, Supplementary Semester and Remedial Exams</td><td>8 weeks</td></tr>
</table>

4.7 Students admitted on transfer from JNTUH affiliated institutes, Universities and other institutes in the subjects in which they are required to earn credits so as to be on par with regular students as prescribed by concerned ‘Board of Studies’.

5.0 REGISTRATION / DROPPING / WITHDRAWAL

The academic calendar includes important academic activities to assist the students and the faculty. These include, dates assigned for registration of courses, dropping of courses and withdrawal from courses. This enables the students to be well prepared and take full advantage of the flexibility provided by the credit system.

5.1. Each student has to compulsorily register for course work at the beginning of each semester as per the schedule mentioned in the Academic Calendar. It is compulsory for the student to register for courses in time. The registration will be organized departmentally under the supervision of the Head of the Department.

5.2. In ABSENTIA, registration will not be permitted under any circumstances.

5.3. At the time of registration, students should have cleared all the dues of Institute and Hostel for the previous semesters, paid the prescribed fees for the current semester and not been debarred from the institute for a specified period on disciplinary or any other ground.

5.4. In the first two semesters, the prescribed course load per semester is fixed and is mandated to registered all courses. Withdrawal / dropping of courses in the first and second semester is not allowed.

5.5. In higher semesters, the average load is 22 credits / semester, with its minimum and maximum limits being set at 16 and 28 credits. This flexibility enables students (**from IV semester onwards**) to cope-up with the course work considering the academic strength and capability of student.

5.6. **Dropping of Courses:**

Within one week after the last date of first internal assessment test or by the date notified in the academic calendar, the student may in consultation with his / her faculty mentor/adviser, drop one or more courses without prejudice to the minimum number of credits as specified in clause 5.4. The dropped courses are not recorded in the memorandum of grades. Student must complete the dropped subject by registering in the supplementary semester / forthcoming semester in order to earn the required credits. Student must complete the dropped subject by registering in the supplementary semester / forthcoming semester in order to earn the required credits.

5.7. **Withdrawal from Courses:**

A student is permitted to withdraw from a course by the date notified in the academic calendar. Such withdrawals will be permitted without prejudice to the minimum number of credits as specified in clause 5.4. A student cannot withdraw a course more than once and withdrawal of reregistered courses is not permitted.

6.0 CREDIT SYSTEM

The B.Tech Program shall consist of a number of courses and each course shall be assigned with credits. The curriculum shall comprise Theory Courses, Elective Courses, Laboratory Courses, Value Added Courses, Mandatory Courses, Experiential Engineering Education (ExEEd), Internship and Project work.

Depending on the complexity and volume of the course, the number of contact periods per week will be assigned. Each theory and laboratory course carries credits based on the number of hours / week.

- Contact classes (Theory): 1 credit per lecture hour per week, 1 credit per tutorial hour per week.
- Laboratory hours (Practical): 1 credit for 2 practical hours per week.
- Project work: 1 credit for 2 hours of project work per week.
- Mandatory Courses: No credit is awarded.
- Value Added Courses: No credit is awarded.
- Experiential Engineering Education (ExEEd): 1 credit for two per hours.

Credit distribution for courses offered is given in Table 5.

Table 5: Credit distribution

S. No	Course	Hours	Credits
1	Theory Course	2 / 3 / 4	2 / 3 / 4
2	Elective Courses	3	3
3	Laboratory Courses	2 / 3 / 4	1 / 1.5 / 2

4	Mandatory Course / Value Added Course	-	0
5	Project Work	-	10
6	Full Semester Internship (FSI) Project work	-	10

Major benefits of adopting the credit system are listed below:

- Quantification and uniformity in the listing of courses for all programs at College, like core, electives and project work.
- Ease of allocation of courses under different heads by using their credits to meet national /international practices in technical education.
- Convenience to specify the minimum / maximum limits of course load and its average per semester in the form of credits to be earned by a student.
- Flexibility in program duration for students by enabling them to pace their course load within minimum/maximum limits based on their preparation and capabilities.
- Wider choice of courses available from any department of the same College or even from other similar Colleges, either for credit or for audit.
- Improved facility for students to optimize their learning by availing of transfer of credits earned by them from one College to another.

7.0 CURRICULAR COMPONENTS

Courses in a curriculum may be of three kinds: **Foundation / Skill, Core and Elective Courses.**

Foundation / Skill Course:

Foundation courses are the courses based upon the content leads to enhancement of skill and knowledge as well as value based and are aimed at man making education. Skill subjects are those areas in which one needs to develop a set of skills to learn anything at all. They are fundamental to learning any subject.

Professional Core Courses:

There may be a core course in every semester. This is the course which is to be compulsorily studied by a student as a core requirement to complete the requirement of a program in the said discipline of study.

Elective Course:

Electives provide breadth of experience in respective branch and application areas. Elective course is a course which can be chosen from a pool of courses. It may be:

- Supportive to the discipline of study
- Providing an expanded scope
- Enabling an exposure to some other discipline / domain
- Nurturing student’s proficiency / skill.

An elective may be Professional Elective, is a discipline centric focusing on those courses which add generic proficiency to the students or may be Open Elective, chosen from unrelated disciplines.

There are six professional elective tracks; students can choose not more than two courses from each track. Overall, students can opt for six professional elective courses which suit their project work in consultation with the faculty advisor/mentor. Nevertheless, one course from each of the four open electives has to be selected. A student may also opt for more elective courses in his/her area of interest.

Every course of the B.Tech program will be placed in one of the eight categories with minimum credits as listed in the Table 6.

Table 6: Category Wise Distribution of Credits

S. No	Category	Breakup of Credits
1	Humanities and Social Sciences (HSMC), including Management.	6
2	Basic Science Courses (BSC) including Mathematics, Physics and Chemistry.	18.5
3	Engineering Science Courses (ESC), including Workshop, Drawing, ExEEd, Basics of Electrical / Electronics / Mechanical / Computer Engineering.	20.5
4	Professional Core Courses (PCC), relevant to the chosen specialization / branch.	78
5	Professional Electives Courses (PEC), relevant to the chosen specialization / branch.	18
6	Open Elective Courses (OEC), from other technical and/or emerging subject areas.	09
7	Project work (PROJ) / Full Semester Internship (FSI) Project work	10
8	Mandatory Courses (MC) / Value Added Courses (VAC).	Non-Credit
TOTAL		**160**

Semester wise course break-up

Following are the **TWO** models of course structure out of which any student shall choose or will be allotted with one model based on their academic performance.

i. Full Semester Internship (FSI) Model and

ii. Non Full Semester Internship (NFSI) Model

In the FSI Model, out of the selected students - half of students shall undergo Full Semester Internship in VII semester and the remaining students in VIII semester. In the Non-FSI Model, all the selected students shall carry out the course work and Project work as specified in the course structure. A student who secures a minimum CGPA of 7.5 upto IV semester with **no current arrears** and maintains the CGPA of 7.5 till VI Semester shall be eligible to opt for FSI.

8. EVALUATION METHODOLOGY

Each theory course will be evaluated for a total of 100 marks, with 30 marks for Continuous Internal Assessment (CIA) and 70 marks for Semester End Examination (SEE). Student's performance in a course shall be judged by taking into account the results of CIA and SEE together. Table-7 shows the typical distribution of weightage for CIA and SEE.

Table 7: Assessment pattern for Theory Courses

Component		Marks	Total Marks
CIA	Continuous Internal Examination – 1 (Mid-term)	10	30
	Continuous Internal Examination – 2 (End-term)	10	
	Tech talk / Quiz – 1 and Quiz – 2	5	
	Concept video / Alternative Assessment Tool (AAT)	5	
SEE	Semester End Examination (SEE)	70	70
Total Marks			**100**

8.1. Semester End Examination (SEE):

The SEE is conducted for 70 marks of 3 hours duration. The syllabus for the theory courses is divided into FIVE modules and each modules carries equal weightage in terms of marks distribution. The question paper pattern is as follows.

Two full questions with 'either' 'or' choice will be drawn from each module. Each question carries 14 marks. There could be a maximum of two sub divisions in a question.

The emphasis on the questions is broadly based on the following criteria:

50 %	To test the objectiveness of the concept
50 %	To test the analytical skill of the concept OR to test the application skill of the concept

8.1. Continuous Internal Assessment (CIA):

For each theory course the CIA shall be conducted by the faculty / teacher handling the course. CIA is conducted for a total of 30 marks, with 20 marks for Continuous Internal Examination (CIE), 05 marks for Quiz and 05 marks for Alternative Assessment Tool (AAT). **Two CIE Tests are Compulsory** and sum of the two tests, along with the scores obtained in the quizzes (average of Quiz – 1 and Quiz – 2) / AAT shall be considered for computing the final CIA of a student in a given course.

The CIE Tests/quizzes/AAT shall be conducted by the course faculty with due approval from the HOD. Advance notification for the conduction of Quiz/AAT is mandatory and the responsibility lies with the concerned course faculty.

8.1.1. Continuous Internal Examination (CIE):

Two CIE exams shall be conducted at the end of the 8th and 16th week of the semester respectively for 10 marks each of 2 hours duration consisting of five descriptive type questions out of which four questions have to be answered. The valuation and verification of answer scripts of CIE exams shall be completed within a week after the conduct of the Examination.

8.1.2. Quiz – Online Examination

Two Quiz exams shall be conducted along with CIE in online mode for 5 marks each, consisting of 10 short answers questions (Definitions and Terminology) and 10 multiple choice questions (having each question to be answered by tick marking the correct answer from the choices (commonly four) given against it. Such a question paper shall be useful in testing of knowledge, skills, application, analysis, evaluation and understanding of the students. Average of two quiz examinations shall be considered.

8.1.3. Alternative Assessment Tool (AAT)

In order to encourage innovative methods while delivering a course, the faculty members are encouraged to use the Alternative Assessment Tool (AAT). This AAT enables faculty to design own assessment patterns during the CIA. The AAT enhances the autonomy (freedom and flexibility) of individual faculty and enables them to create innovative pedagogical practices. If properly applied, the AAT converts the classroom into an effective learning centre.
The AAT may include tech talk, tutorial hours/classes, seminars, assignments, term paper, open ended experiments, concept videos, partial reproduction of research work, oral presentation of research work, developing a generic tool-box for problem solving, report based on participation in create-a-thon, make-a-thon, code-a-thon, hack-a-thon conducted by reputed organizations / any other. etc.

However, it is mandatory for a faculty to obtain prior permission from the concerned HOD and spell out the teaching/assessment pattern of the AAT prior to commencement of the classes.

8.2 Laboratory Course

Each laboratory will be evaluated for a total of 100 marks consisting of 30 marks for internal assessment and 70 marks for semester end lab examination. Out of 30 marks of internal assessment, continuous lab assessment will be done for 20 marks for the day to day performance and 10 marks for the final internal lab assessment. The semester end laboratory examination for 70 marks shall be conducted internally by the respective department with at least two faculty members as examiners, both nominated by the Principal from the panel of experts recommended by the Chairman, BOS.

All the drawing related courses are evaluated in line with laboratory courses. The distribution shall be 30 marks for internal evaluation (20 marks for day–to–day work, and 10 marks for internal tests) and 70 marks for semester end laboratory examination. There shall be ONE internal test of 10 marks in each semester.

8.3 Audit Courses

In Addition, a student can register for courses for audit only with a view to supplement his/her knowledge and/or skills. Here also, the student's grades shall have to be reflected in the Memorandum of Grades. But, these shall not be taken into account in determining the student's academic performance in the semester. In view of this, it shall not be necessary for the institute to issue any separate transcript covering the audit courses to the registrants at these courses. Its result shall be declared as "Satisfactory" or "Not Satisfactory" performance.

8.4 Mandatory Courses (MC)

These courses are among the compulsory courses but will not carry any credits. However, a pass in each such course during the program shall be necessary requirement for the student to qualify for the award of Degree. Its result shall be declared as "Satisfactory" or "Not Satisfactory" performance.

8.5 Additional Mandatory Courses for lateral entry B.Tech students

In addition to the non-credit mandatory courses for regular B.Tech students, the lateral entry students shall take up the following three non-credit mandatory bridge courses (one in III semester, one in IV semester and one in V semester) as listed in Table 8. The student shall pass the following non-credit mandatory courses for the award of the degree and must clear these bridge courses before advancing to the VII semester of the program.

Table-8: Additional Mandatory Courses for lateral entry

S. No	Additional mandatory courses for lateral entry students
1	Dip-Mathematics
2	Dip-Programming for Problem Solving
3	Dip-English Communication Skills

8.6 Value Added Courses

The value added courses are audit courses offered through joint ventures with various organizations providing ample scope for the students as well as faculty to keep pace with the latest technologies pertaining to their chosen fields of study. A plenty of value added programs will be proposed by the departments one week before the commencement of class work. The students are given the option to choose the courses according to their desires and inclinations as they choose the desired items in a cafeteria. The expertise gained through the value added programs should enable them to face the formidable challenges of the future and also assist them in exploring new opportunities. Its result shall be declared with "Satisfactory" or "Not Satisfactory" performance.

8.7 Experiential Engineering Education (ExEED)

Engineering entrepreneurship requires strong technical skills in engineering design and computation with key business skills from marketing to business model generation. Students require sufficient skills to innovate in existing companies or create their own.

This course will be evaluated for a total of 100 marks consisting of 30 marks for internal assessment and 70 marks for semester end Examination. Out of 30 marks of internal assessment, The Student has to submit Innovative Idea in a team of four members in the given format. The semester end examination for 70 marks shall be conducted internally, students has to present the Innovative Idea and it will be evaluated by internal ExEEd faculty with at least one faculty member as examiner from the industry, both nominated by the Principal from the panel of experts recommended by the Dean-CLET.

8.8 Project Work / FSI Project Work

This gives students a platform to experience a research driven career in engineering, while developing a device / systems and publishing in reputed SCI / SCOPUS indexed journals and/or filing an **Intellectual Property** (IPR-Patent/Copyright) to aid communities around the world. Students should work individually as per the guidelines issued by head of the department concerned. The benefits to students of this mode of learning include increased engagement, fostering of critical thinking and greater independence.

The topic should be so selected that the students are enabled to complete the work in the stipulated time with the available resources in the respective laboratories. The scope of the work be handling part of the consultancy work, maintenance of the existing equipment, development of new experiment setup or can be a prelude to the main project with a specific outcome.

Project report will be evaluated for 100 marks in total. Assessment will be done for 100 marks out of which, the supervisor / guide will evaluate for 30 marks based on the work and presentation / execution of the work. Subdivision for the remaining 70 marks is based on publication, report, presentation, execution and viva-voce. Evaluation shall be done by a committee comprising the supervisor, Head of the department and an examiner nominated by the Principal from the panel of experts recommended by Chairman, BOS in consultation with Head of the department.

8.8.1 Project work

The student's project activity is spread over in VII semester and in VIII semesters. A student shall carry out the project work under the supervision of a faculty member or in collaboration with an Industry, R&D organization or another academic institution/University where sufficient facilities exist to carry out the project work.

Project work (phase-I) starts in VII semester as it takes a vital role in campus hiring process. Students shall select project titles from their respective logins uploaded by the supervisors at the beginning of VII semester. Three reviews are conducted by department review committee (DRC) for 10 marks each. Student must submit a project report summarizing the work done up to design phase/prototype by the end of VII semester. The semester end examination for project work (phase-I) is evaluated based on the project report submitted and a viva-voce exam for 70 marks by a committee comprising the head of the department, the project supervisor and an external examiner nominated by the Principal.

Project Work (phase-II) starts in VIII semester, shall be evaluated for 100 marks out of which 30 marks towards continuous internal assessment and 70 marks for semester end examination. Three reviews are to be conducted by DRC on the progress of the project for 30 marks. The semester end examination shall be based on the final report submitted and a viva-voce exam for 70 marks by a committee comprising the head of the department, the project supervisor and an external examiner nominated by the Principal.

A minimum of 40% of maximum marks shall be obtained to earn the corresponding credits.

8.8.2 Full Semester Internship (FSI)

FSI is a full semester internship program carry 10 credits. The FSI shall be opted in VII semester or in VIII semester. During the FSI, student has to spend one full semester in an identified industry / firm / R&D organization or another academic institution/University where sufficient facilities exist to carry out the project work.

Following are the evaluation guidelines:

- Quizzes: 2 times
- Quiz #1 - About the industry profile, weightage: 5%
- Quiz #2 - Technical-project related, weightage: 5%
- Seminars - 2 times (once in six weeks), weightage: 7.5% + 7.5%
- Viva-voce: 2 times (once in six weeks), weightage: 7.5% + 7.5%
- Project Report, weightage: 15%
- Internship Diary, weightage: 5 %
- Final Presentation, weightage: 40%

FSI shall be open to all the branches with a ceiling of maximum 10% distributed in both semesters. The selection procedure is:

- Choice of the students.
- CGPA (> 7.5) upto IV semester having no credit arrears.
- Competency Mapping / Allotment.

It is recommended that the FSI Project work leads to a research publication in a reputed Journal/Conference or the filing of patent/design with the patent office, or, the start-up initiative with a sustainable and viable business model accepted by the incubation center of the institute together with the formal registration of the startup.

8.9 Plagiarism index for Project Report:

All project reports shall go through the plagiarism check and the plagiarism index has to be less than 20%. Project reports with plagiarism more than 20% and less than 60% shall be asked for resubmission within a stipulated period of six months. Project reports with plagiarism more than 60% shall be rejected.

9. MAKEUP EXAMINATION

The make-up examination facility shall be available to students who may have missed to attend **CIE/Quiz** of one or more courses in a semester for valid reasons. The CIE make-up examination shall have comprehensive online objective type questions for 20 marks and Quiz for 5 marks. The content for the make-up examination shall be on the whole syllabus. The Makeup examination shall be conducted at the end of the respective semester.

10. SUPPLEMENTARY EXAMINATIONS

In addition to the Regular Semester End Examinations held at the end of each semester, Supplementary Semester End Examinations will be conducted within three weeks of the commencement of the teaching of the next semester. Candidates taking the Regular / Supplementary examinations as Supplementary candidates may have to take more than one Semester End Examination per day. A student can appear for any number of supplementary examinations till he/she clears all courses which he/she could not clear in the first attempt. However the maximum stipulated period for the course shall not be relaxed under any circumstances.

11. ATTENDANCE REQUIREMENTS AND DETENTION POLICY

11.1 It is desirable for a candidate to have 100% attendance in each course. In every course (theory/laboratory), student has to maintain a minimum of 75% attendance including the days of attendance in sports, games, NCC and NSS activities to be eligible for appearing in Semester End Examination of the course.

11.2 In case of medical issues, deficiency of attendance in each course to the extent of 10% may be condoned by the College Academic Committee (CAC) on the recommendation of the Head of the Department if the attendance is between 75% and 65% in every course, subjected to the submission of medical certificates, medical case file, and other needful documents to the concerned departments.

11.3 The basis for the calculation of the attendance shall be the period prescribed by the institute by its calendar of events. For late admission, attendance is reckoned from the date of admission to the program. However, in case of a student having less than 65% attendance in any course, s/he shall be detained in the course and in no case such process will be relaxed.

11.4 A candidate shall put in a minimum required attendance in atleast 60% of (rounded to the next highest integer) theory courses for getting promoted to next higher class / semester. Otherwise, s/he shall be declared detained and has to repeat semester.

11.5 Students whose shortage of attendance is not condoned in any subject are not eligible to write their semester end examination of that courses and their registration shall stand cancelled.

11.6 A prescribed fee shall be payable towards condonation of shortage of attendance.

11.7 A student shall not be promoted to the next semester unless he satisfies the attendance requirement of the present semester, as applicable. They may seek readmission into that semester when offered next. If any candidate fails to fulfill the attendance requirement in the present semester, he shall not be eligible for readmission into the same class.

11.8 Any student against whom any disciplinary action by the institute is pending shall not be permitted to attend any SEE in that semester.

12. CONDUCT OF SEMESTER END EXAMINATIONS AND EVALUATION

12.1 Semester end examination shall be conducted by the Controller of Examinations (COE) by inviting Question Papers from the External Examiners.

12.2 Question papers may be moderated for the coverage of syllabus, pattern of questions by a Semester End Examination Committee chaired by Head of the Department one day before the commencement of semester end examinations. Internal Examiner shall prepare a detailed scheme of valuation.

12.3 The answer papers of semester end examination should be evaluated by the internal examiner immediately after the completion of exam and the award sheet should be submitted to COE in a sealed cover.

12.4 COE shall invite 3 - 9 internal/external examiners to evaluate all the semester end examination answer books on a prescribed date(s). Practical laboratory exams are conducted involving external examiners.

12.5 Examinations Control Committee shall consolidate the marks awarded by examiner/s and award grades.

13. SCHEME FOR THE AWARD OF GRADE

13.1 A student shall be deemed to have satisfied the minimum academic requirements and earn the credits for each theory course, if s/he secures

a) Not less than 35% marks for each theory course in the semester end examination, and

b) A minimum of 40% marks for each theory course considering Continuous Internal Assessment (CIA) and Semester End Examination (SEE).

13.2 A student shall be deemed to have satisfied the minimum academic requirements and earn the credits for each Laboratory / Project work / FSI Project work, if s/he secures

a) Not less than 40% marks for each Laboratory / Project work / FSI Project work course in the semester end examination,

b) A minimum of 40% marks for each Laboratory / Project work / FSI Project work course considering both internal and semester end examination.

13.3 If a candidate fails to secure a pass in a particular course, it is mandatory that s/he shall register and reappear for the examination in that course during the next semester when examination is conducted in that course. It is mandatory that s/he should continue to register and reappear for the examination till s/he secures a pass.

13.4 A student shall be declared successful or 'passed' in a semester, if he secures a Grade Point ≥ 5 ('C' grade or above) in every course in that semester (i.e. when the student gets an SGPA ≥5.0 at the end of that particular semester); and he shall be declared successful or 'passed' in the entire under graduate programme, only when gets a CGPA ≥5.0 for the award of the degree as required.

14. LETTER GRADES AND GRADE POINTS

14.1 Performances of students in each course are expressed in terms of marks as well as in Letter Grades based on absolute grading system. The UGC recommends a 10-point grading system with the following letter grades as given in the Table-9.

Table-9: Grade Points Scale (Absolute Grading)

Range of Marks	Grade Point	Letter Grade
100 – 90	10	S (Superior)
89 – 80	9	A+ (Excellent)
79 – 70	8	A (Very Good)
69 – 60	7	B+ (Good)
59 – 50	6	B (Average)
49 – 40	5	C (Pass)
Below 40	0	F (Fail)
Absent	0	AB (Absent)
Authorized Break of Study	0	ABS

14.2 A student is deemed to have passed and acquired to correspondent credits in particular course if s/he obtains any one of the following grades: “S”, “A+”, “A”, “B+”, “B”, “C”.

14.3 A student obtaining Grade F shall be considered Failed and will be required to reappear in the examination.

14.4 For non credit courses, ‘Satisfactory’ or “Not Satisfactory” is indicated instead of the letter grade and this will not be counted for the computation of SGPA/CGPA.

14.5 “SA” denotes shortage of attendance (as per item 11) and hence prevention from writing Semester End Examination.

14.6 “W” denotes **withdrawal** from the exam for the particular course.

14.7 At the end of each semester, the institute issues grade sheet indicating the SGPA and CGPA of the student. However, grade sheet will not be issued to the student if s/he has any outstanding dues.

14.8 **Award of Class:**

Sometimes, it is necessary to provide equivalence of these averages, viz., SGPA and CGPA with the percentages and/or Class awarded as in the conventional system of declaring the results of University examinations. This shall be done by Autonomous Colleges under the University only at one stage by prescribing certain specific thresholds in these averages for First Class with Distinction, First Class and Second Class, at the time of Degree Award. This provision given in Table-10 follows the approach of the Council for this purpose as reproduced from the AICTE Approval Process Handbook:

Table 10: Percentage Equivalence of Grade Points (for a 10 – Point Scale)

Grade Point	Percentage of Marks / Class
5.5	50
6.0	55
6.5	60
7.0	65
7.5	70
8.0	75

Note:

(1) The following Formula for Conversion of CGPA to percentage of marks to be used only after a student has successfully completed the program:
Percentage of Marks = $(CGPA - 0.5) \times 10$

(2) Class designation:
≥75% (First Class with Distinction),
≥ 60% and <75 % (First Class),
≥ 50 % and <60% (Second Class),
≥45% and <50% (Pass Class).

(3) The SGPA will be computed and printed on the Memorandum of Grades only if the candidate passes in all the courses offered and gets minimum B grade in all the courses.

(4) CGPA is calculated only when the candidate passes in all the courses offered in all the semesters.

15. COMPUTATION OF SGPA AND CGPA

The UGC recommends to compute the Semester Grade Point Average (SGPA) and Cumulative Grade Point Average (CGPA). The credit points earned by a student are used for calculating the Semester Grade Point Average (SGPA) and the Cumulative Grade Point Average (CGPA), both of which are important performance indices of the student. SGPA is equal to the sum of all the total points earned by the student in a given semester divided by the number of credits registered by the student in that semester. CGPA gives the sum of all the total points earned in all the previous semesters and the current semester divided by the number of credits registered in all these semesters. Thus,

$$SGPA = \sum_{i=1}^{n}(C_i G_i) / \sum_{i=1}^{n} C_i$$

Where, C_i is the number of credits of the i^{th} course and G_i is the grade point scored by the student in the i^{th} course and n represent the number of courses in which a student is registered in the concerned semester.

$$CGPA = \sum_{j=1}^{m}(C_j S_j) / \sum_{j=1}^{m} C_j$$

Where, S_j is the SGPA of the j^{th} semester and C_j is the total number of credits upto the semester and m represent the number of semesters completed in which a student registered upto the semester.

The SGPA and CGPA shall be rounded off to 2 decimal points and reported in the transcripts.

16. ILLUSTRATION OF COMPUTATION OF SGPA AND CGPA

16.1 Illustration for SGPA

Course Name	Course Credits	Grade letter	Grade point	Credit Point (Credit x Grade)
Course 1	3	A	8	3 x 8 = 24
Course 2	4	B+	7	4 x 7 = 28
Course 3	3	B	6	3 x 6 = 18
Course 4	3	S	10	3 x 10 = 30
Course 5	3	C	5	3 x 5 = 15
Course 6	4	B	6	4 x 6 = 24
	20			**139**

Thus, SGPA = 139 / 20 = 6.95

16.2 Illustration for CGPA

Semester 1	Semester 2	Semester 3	Semester 4
Credit: 20 SGPA: 6.9	Credit: 22 SGPA: 7.8	Credit: 25 SGPA: 5.6	Credit: 26 SGPA: 6.0
Semester 5	**Semester 6**		
Credit: 26 SGPA: 6.3	Credit: 25 SGPA: 8.0		

$$Thus,\ CGPA = \frac{20x6.9\ +\ 22x7.8\ +\ 25x5.6\ +\ 26x6.0\ +\ 26x6.3\ +\ 25x8.0}{144} = 6.73$$

17. REVIEW OF SEE THEORY ANSWER BOOKS

Semester end examination answer books are made available online in CMS portal on the day of publication of results. A student, who is not satisfied with the assessment, is directed to apply for the review of his/her semester end examination answer book(s) in the theory course(s), within 2 working days from the publication of results in the prescribed format to the Controller of Examinations through the Head of the department with prescribed fee.

The Controller of Examinations shall appoint two examiners (chief examiner of original exam and a new examiner) for the review of the semester end examination (theory) answer book. Both examiners shall jointly review and marks awarded in the previous assessment shall be kept open.

The marks obtained by the candidate after the review shall be considered for grading, only if, the change in mark is more than or equal to 10% of total mark of semester end examination (theory). Marks obtained after re-evaluation shall stand final even if it is less than the original marks. Review is not permitted to the courses other than theory courses.

18. PROMOTION POLICIES

The following academic requirements have to be satisfied in addition to the attendance requirements mentioned in item no. 11.

18.1 For students admitted into B.Tech (Regular) program

18.1.1 A student will not be promoted from II semester to III semester unless s/he fulfills the academic requirement of securing 50% of the total credits (rounded to the next lowest integer) from I and II semester examinations, whether the candidate takes the examination(s) or not.

18.1.2 A student will not be promoted from IV semester to V semester unless s/he fulfills the academic requirement of securing 60% of the total credits (rounded to the next lowest integer) upto III semester **or** 60% of the total credits (rounded to the next lowest integer) up to IV semester, from all the examinations, whether the candidate takes the examination(s) or not.

18.1.3 A student shall be promoted from VI semester to VII semester only if s/he fulfills the academic requirements of securing 60% of the total credits (rounded to the next lowest integer) up to V semester **or** 60% of the total credits (rounded to the next lowest integer) up to VI semester from all the examinations, whether the candidate takes the examination(s) or not.

18.1.4 A student shall register for all the 160 credits and earn all the 160 credits. Marks obtained in all the 160 credits shall be considered for the award of the Grade.

18.2 For students admitted into B.Tech (lateral entry students)

18.2.1 A student will not be promoted from IV semester to V semester unless s/he fulfills the academic requirement of securing 60% of the total credits (rounded to the next lowest integer) up to IV semester, from all the examinations, whether the candidate takes the examination(s) or not.

18.2.2 A student shall be promoted from VI semester to VII semester only if s/he fulfills the academic requirements of securing 60% of the total credits (rounded to the next lowest integer) up to V semester **or** 60% of the total credits (rounded to the next lowest integer) up to VI semester from all the examinations, whether the candidate takes the examination(s) or not.

18.2.3 A student shall register for all the 126 credits and earn all the 126 credits. Marks obtained in all the 126 credits shall be considered for the award of the Grade.

19. GRADUATION REQUIREMENTS

The following academic requirements shall be met for the award of the B.Tech degree.

19.1 Student shall register and acquire minimum attendance in all courses and secure 160 credits (with minimum CGPA of 5.0), for regular program and 126 credits (with minimum CGPA of 5.0), for lateral entry program.

19.2 A student of a regular program, who fails to earn 160 credits within eight consecutive academic years from the year of his/her admission with a minimum CGPA of 5.0, shall forfeit his/her degree and his/her admission stands cancelled.

19.3 A student of a lateral entry program who fails to earn 126 credits within six consecutive academic years from the year of his/her admission with a minimum CGPA of 5.0, shall forfeit his/her degree and his/her admission stands cancelled.

20. BETTERMENT OF MARKS IN THE COURSES ALREADY PASSED

Students who clear all the courses in their first attempt and wish to improve their CGPA shall register and appear for betterment of marks for one course of any theory courses within a period of subsequent two semesters. The improved marks shall be considered for classification / distinction but not for ranking. If there is no improvement, there shall not be any change in the original marks already awarded.

21. AWARD OF DEGREE

21.1 Classification of degree will be as follows:

CGPA ≥ 8.0	CGPA ≥ 6.5 and < 8.0	CGPA ≥ 5.5 and < 6.5	CGPA ≥ 5.0 and < 5.5	CGPA < 5.0
First Class with Distinction	**First Class**	**Second Class**	**Pass Class**	**Fail**

21.2 A student with final CGPA (at the end of the under graduate programme) ≥8.00, and fulfilling the following conditions - shall be placed in '**first class with distinction**'. However,

(a) Should have passed all the courses in '**first appearance'** within the first 4 academic years (or 8 sequential semesters) from the date of commencement of first year first semester.

(b) Should have secured a CGPA ≥8.00, at the end of each of the 8 sequential semesters, starting from I year I semester onwards.

(c) Should not have been detained or prevented from writing the semester end examinations in any semester due to shortage of attendance or any other reason.

A student not fulfilling any of the above conditions with final CGPA >8 shall be placed in **'first class'.**

21.3 Students with final CGPA (at the end of the B.Tech program) ≥6.50 but <8.00 shall be placed in **'first class'.**

21.4 Students with final CGPA (at the end of the B.Tech program) ≥5.50 but <6.50, shall be placed in '**second class'**.

21.5 All other students who qualify for the award of the degree (as per item 19), with final CGPA (at the end of the B.Tech program) ≥5.0 but <5.50, shall be placed in '**pass class**'.

21.6 A student with final CGPA (at the end of the B.Tech program) < 5.00 will not be eligible for the award of the degree.

21.7 Students fulfilling the conditions listed under item 21.2 alone will be eligible for award of '**Gold Medal**'.

21.8. In order to extend the benefit to the students with one/two backlogs after either VI semester or VIII semester, GRAFTING option is provided to the students enabling their placements and fulfilling graduation requirements. Following are the guidelines for the Grafting:

(a) Grafting will be done among the courses within the semester shall draw a maximum of 7 marks from the any one of the cleared courses in the semester and will be grafted to the failed course in the same semester.

(b) Students shall be given a choice of grafting only once in the 4 years program, either after VI semester (Option #1) or after VIII semester (Option #2).

(c) Option#1: Applicable to students who have maximum of TWO theory courses in V and / or VI semesters.

Option#2: Applicable to students who have maximum of TWO theory courses in VII and / or VIII semesters.

(d) Eligibility for grafting:

i. Prior to the conduct of the supplementary examination after the declaration of VI or VIII semester results.

ii. S/he must appear in all regular or supplementary examinations as per the provisions laid down in regulations for the courses s/he appeals for grafting.

iii. The marks obtained by her/him in latest attempt shall be taken into account for grafting of marks in the failed course(s).

21.9 Student, who clears all the courses upto VII semester, shall have a chance to appear for Quick Supplementary Examination to clear the failed courses of VIII semester.

21.10 By the end of VI semester, all the students (regular and lateral entry students) shall complete one of the Value added course and mandatory course with acceptable performance.

21.11 In case, a student takes more than one attempt in clearing a course, the final marks secured shall be indicated by * mark in the grade sheet.

All the candidates who register for the semester end examination will be issued a memorandum of grades sheet by the institute. Apart from the semester wise memorandum of grades sheet, the institute will issue the provisional certificate and consolidated grades memorandum subject to the fulfillment of all the academic requirements.

22. B.TECH WITH HONOURS OR ADDITIONAL MINORS IN ENGINEERING

Students acquiring 160 credits are eligible to get B.Tech degree in Engineering. A student will be eligible to get B.Tech degree with Honours or additional Minors in Engineering, if s/he completes an additional 20 credits (3/4 credits per course). These could be acquired through MOOCs from SWAYAM / NPTEL / edX / Coursera / Udacity / PurdueNext / Khan Academy / QEEE etc. The list for MOOCs will be a dynamic one, as new courses are added from time to time. Few essential skill sets required for employability are also identified year wise. Students interested in doing MOOC courses shall register the course title at their department office at the start of the semester against the courses that are announced by the department. Any expense incurred for the MOOC course / summer program should be met by the students.

Only students having no credit arrears and a CGPA of 7.5 or above at the end of the fourth semester are eligible to register for B.Tech (Honours / Minor). After registering for the B.Tech (Honours / Minor) program, if a student fails in any course, s/he will not be eligible for B.Tech (Honours / Minor).

Every Department to develop and submit a Honours / Minors – courses list of 5 - 6 theory courses.

Honours Certificate for Vertical in his/her OWN Branch for Research orientation; Minor in any other branch for Improving Employability.

For the MOOCs platforms, where examination or assessment is absent (like SWAYAM) or where certification is costly (like Coursera or edX), faculty members of the institute prepare the examination question papers, for the courses undertaken by the students of respective Institutes, so that examinations Control Office (ECO) can conduct examination for the course. There shall be one Continuous Internal Examination (Quiz exam for 30 marks) after 8 weeks of the commencement of the course and semester end examination (Descriptive exam for 70 marks) shall be done along with the other regular courses.

A student can enroll for both Minor & Honours or for two Minors. The final grade sheet will only show the basic CGPA corresponding to the minimum requirement for the degree. The Minors/Honours will be indicated by a separate CGPA. The additional courses taken will also find separate mention in the grade sheet.

If a student drops (or terminated) from the Minor/Honours program, they cannot convert the earned credits into free or core electives; they will remain extra. These additional courses will find mention in the grade sheet (but not in the degree certificate). In such cases, the student may choose between the actual grade or a "Pass (P)" grade and also choose to omit the mention of the course as for the following:

- All the courses done under the dropped Minor/Honours will be shown in the grade sheet
- None of the courses done under the dropped Minor/Honours will be shown in the grade sheet.

Honours will be reflected in the degree certificate as "B.Tech (Honours) in XYZ Engineering". Similarly, Minor as "B.Tech in XYZ Engineering with Minor in ABC". If a student has done both Honours & Minor, it will be acknowledged as "B.Tech (Honours) in XYZ Engineering with Minor in ABC". And two minors will be reflected as "B.Tech in XYZ Engineering with Minor in ABC and Minor in DEF".

22.1. B.Tech with Honours

The total of 20 credits required to be attained for B.Tech Honours degree are distributed from V semester to VII semester in the following way:

For V semester	:	4 – 8 credits
For VI semester	:	4 – 8 credits
For VII semester	:	4 – 8 credits

Following are the details of such Honours which include some of the most interesting areas in the profession today:

S. No	Department	Honours scheme
1	Aeronautical Engineering	Aerospace Engineering / Space Science etc.
2	Computer Science and Engineering / Information Technology	Big data and Analytics / Cyber Physical Systems, Information Security / Cognitive Science / Artificial Intelligence/ Machine Learning / Data Science / Internet of Things (IoT) etc.
3	Electronics and Communication Engineering	Digital Communication / Signal Processing / Communication Networks / VLSI Design / Embedded Systems etc.
4	Electrical and Electronics Engineering	Renewable Energy systems / Energy and Sustainability / IoT Applications in Green Energy Systems etc.
5	Mechanical Engineering	Industrial Automation and Robotics / Manufacturing Sciences and Computation Techniques etc.
6	Civil Engineering	Structural Engineering / Environmental Engineering etc.

22.2 B.Tech with additional Minor in Engineering

Every department to develop and submit Minor courses list of 5 - 6 Theory courses. Student from any department is eligible to apply for Minor from any other department. The total of 20 credits to complete the B.Tech (Minor) program by registering for MOOC courses each having a minimum of 3/4 credits offered by reputed institutions / organization with the approval of the department. Registration of the student for B.Tech (Minor), is from V Semester to VII Semester of the program in the following way:

For V semester	:	4 – 8 credits
For VI semester	:	4 – 8 credits
For VII semester	:	4 – 8 credits

Only students having no credit arrears and a CGPA of 7.5 or above at the end of the fourth semester are eligible to register for B.Tech (Minor). After registering for the B.Tech (Minor) program, if a student fails in any course, s/he will not be eligible for B.Tech (Minor).

Every student shall also have the option to do a minor in engineering. A major is a primary focus of study and a minor is a secondary focus of study. The minor has to be a subject offered by a department other than the department that offers the major of the student or it can be a different major offered by the same department. For example, a student with the declared major in Computer Science and Engineering (CSE) may opt to do a minor in Physics; in which case, the student shall receive the degree B.Tech, Computer Science and Engineering with a minor in Physics. A student can do Majors in chosen filed as per the career goal, and a minor may be chosen to enhance the major thus adding the diversity, breadth and enhanced skills in the field.

22.3 Advantages of Minor in Engineering:

The minors mentioned above are having lots of advantages and a few are listed below:

1. To apply the inter-disciplinary knowledge gained through a Major (Stream) + Minor.
2. To enable students to pursue allied academic interest in contemporary areas.
3. To provide an academic mechanism for fulfilling multidisciplinary demands of industries.
4. To provide effective yet flexible options for students to achieve basic to intermediate level competence in the Minor area.
5. Provides an opportunity to students to become entrepreneurs and leaders by taking business/ management minor.
6. Combination in the diverse fields of engineering e.g., CSE (Major) + Electronics (Minor) combination increases placement prospects in chip designing companies.
7. Provides an opportunity to Applicants to pursue higher studies in an inter-disciplinary field of study.
8. Provides opportunity to the Applicants to pursue interdisciplinary research.
9. To increase the overall scope of the undergraduate degrees.

22.4 Following are the details of such Minor / Honours which include some of the most interesting areas in the profession today:

1. Aerospace Engineering
2. Space Science
3. Industrial Automation and Robotics
4. Computer Science and Engineering
5. Data Analytics
6. Machine Learning
7. Data Science
8. Artificial Intelligence
9. Information Security
10. Internet of Things
11. Cyber Physical Systems
12. Electronic System Design
13. Renewable Energy Sources
14. Energy and Sustainability
15. Manufacturing Sciences and Computation Techniques
16. Structural Engineering
17. Environmental Engineering
18. Technological Entrepreneurship
19. Materials Engineering
20. Physics (Materials / Nuclear / Optical / Medical)
21. Mathematics (Combinatorics / Logic / Number theory / Dynamical systems and differential equations/ Mathematical physics / Statistics and Probability).

23.0 TEMPORARY BREAK OF STUDY FROM THE PROGRAM

23.1 A candidate is normally not permitted to take a break from the study. However, if a candidate intends to temporarily discontinue the program in the middle for valid reasons (such as accident or hospitalization due to prolonged ill health) and to rejoin the program in a later respective semester, s/he shall seek the approval from the Principal in advance. Such application shall be submitted before the last date for payment of examination fee of the semester in question and forwarded through the Head of the Department stating the reasons for such withdrawal together with supporting documents and endorsement of his / her parent / guardian.

23.2 The institute shall examine such an application and if it finds the case to be genuine, it may permit the student to temporarily withdraw from the program. Such permission is accorded only to those who do not have any outstanding dues / demand at the College / University level including tuition fees, any other fees, library materials etc.

23.3 The candidate has to rejoin the program after the break from the commencement of the respective semester as and when it is offered.

23.4 The total period for completion of the program reckoned from the commencement of the semester to which the candidate was first admitted shall not exceed the maximum period specified in clause 19. The maximum period includes the break period.

23.5 If any candidate is detained for any reason, the period of detention shall not be considered as 'Break of Study'.

24. TERMINATION FROM THE PROGRAM

The admission of a student to the program may be terminated and the student is asked to leave the institute in the following circumstances:

a. The student fails to satisfy the requirements of the program within the maximum period stipulated for that program.

b. A student shall not be permitted to study any semester more than three times during the entire program of study.

c. The student fails to satisfy the norms of discipline specified by the institute from time to time.

25. TRANSCRIPT

The Transcript will be issued to the student as and when required and will contain a consolidated record of all the courses undergone by him/her, grades obtained and CGPA upto the date of issue of transcript. Only last letter grade obtained in a course by the student upto the date of issue of transcript will be shown in the Transcript.

26. WITH-HOLDING OF RESULTS

If the candidate has not paid any dues to the institute / if any case of indiscipline / malpractice is pending against him, the results and the degree of the candidate will be withheld.

27. GRADUATION DAY

The institute shall have its own annual Graduation Day for the award of degrees to the students completing the prescribed academic requirements in each case, in consultation with the University and by following the provisions in the Statute. The college shall institute prizes and medals to meritorious students and award them annually at the Graduation Day. This will greatly encourage the students to strive for excellence in their academic work.

28. DISCIPLINE

Every student is required to observe discipline and decorum both inside and outside the institute and are expected not to indulge in any activity which will tend to bring down the honour of the institute. If a student indulges in malpractice in any of the theory / practical examination, continuous assessment examinations, he/she shall be liable for punitive action as prescribed by the institute from time to time.

29. GRIEVANCE REDRESSAL COMMITTEE

The institute shall form a Grievance Redressal Committee for each course in each department with the Course Teacher and the HOD as the members. This Committee shall solve all grievances related to the course under consideration.

30. TRANSITORY REGULATIONS

A candidate, who is detained or has discontinued a semester, on readmission shall be required to do all the courses in the curriculum prescribed for the batch of students in which the student joins subsequently. However, exemption will be given to those candidates who have already passed such courses in the earlier semester(s) he was originally admitted into and substitute subjects are offered in place of them as decided by the Board of Studies. However, the decision of the Board of Studies will be final.

a) Four Year B.Tech Regular course:

A student who is following Jawaharlal Nehru Technological University (JNTUH) curriculum and detained due to the shortage of attendance at the end of the first semester shall join the autonomous batch of first semester. Such students shall study all the courses prescribed for the batch in which the student joins and considered on par with regular candidates of Autonomous stream and will be governed by the autonomous regulations.

A student who is following JNTUH curriculum, detained due to lack of credits or shortage of attendance at the end of the second semester or at the subsequent semesters shall join with the autonomous batch in the appropriate semester. Such candidates shall be required to pass in all the courses in the program prescribed by the Board of Studies concerned for that batch of students from that semester onwards to be eligible for the award of degree. However, exemption will be given in the courses of the semester(s) of the batch which he had passed earlier and substitute courses will be offered in place of them as decided by the Board of Studies. The student has to clear all his backlog courses up to previous semester by appearing for the supplementary examinations conducted by JNTUH for the award of degree. The total number of credits to be secured for the award of the degree will be sum of the credits up to previous semester under JNTUH regulations and the credits prescribed for the semester in which a candidate seeks readmission and subsequent semesters under the autonomous stream. The class will be awarded based on the academic performance of a student in the autonomous pattern.

b) Three Year B.Tech program under Lateral Entry Scheme:

A student who is following JNTUH curriculum and detained due to the shortage of attendance at the end of the first semester of second year shall join the autonomous batch of third semester. Such students shall study all the courses prescribed for the batch in which the student joins and considered on par with Lateral Entry regular candidates of Autonomous stream and will be governed by the autonomous regulations.

A student who is following JNTUH curriculum, if detained due to lack of credits or shortage of attendance at the end of the second semester of second year or at the subsequent semesters shall join with the autonomous batch in the appropriate semester. Such candidates shall be required to pass in all the courses in the program prescribed by the Board of Studies concerned for that batch of students from that semester onwards to be eligible for the award of degree. However, exemption will be given in the courses of the semester(s) of the batch which he had passed earlier and substitute courses are offered in place of them as decided by the Board of Studies. The student has to clear all his backlog courses up to previous semester by appearing for the supplementary examinations conducted by JNTUH for the award of degree. The total number of credits to be secured for the award of the degree will be sum of the credits up to previous semester under JNTUH regulations and the credits prescribed for the semester in which a candidate seeks readmission and subsequent semesters under the autonomous status. The class will be awarded based on the academic performance of a student in the autonomous pattern.

c) Transfer candidates (from non-autonomous college affiliated to JNTUH):

A student who is following JNTUH curriculum, transferred from other college to this institute in third semester or subsequent semesters shall join with the autonomous batch in the appropriate semester. Such candidates shall be required to pass in all the courses in the program prescribed by the Board of Studies concerned for that batch of students from that semester onwards to be eligible for the award of degree. However, exemption will be given in the courses of the semester(s) of the batch which he had passed earlier and substitute courses are offered in their place as decided by the Board of Studies. The student has to clear all his backlog courses up to previous semester by appearing for the supplementary examinations conducted by JNTUH for the award of degree. The total number of credits to be secured for the award of the degree will be the sum of the credits up to the previous semester under JNTUH regulations and the credits prescribed for the semester in which a candidate joined after transfer and subsequent semesters under the autonomous status. The class will be awarded based on the academic performance of a student in the autonomous pattern.

d) Transfer candidates (from an autonomous college affiliated to JNTUH):

A student who has secured the required credits up to previous semesters as per the regulations of other autonomous institutions shall also be permitted to be transferred to this institute. A student who is transferred from the other autonomous colleges to this institute in third semester or subsequent semesters shall join with the autonomous batch in the appropriate semester. Such candidates shall be required to pass in all the courses in the program prescribed by the Board of Studies concerned for that batch of students from that semester onwards to be eligible for the award of degree. However, exemption will be given in the courses of the semester(s) of the batch which he had passed earlier and substitute subjects are offered in their place as decided by the Board of Studies. The total number of credits to be secured for the award of the degree will be the sum of the credits up to previous semester as per the regulations of the college from which he is transferred and the credits prescribed for the semester in which a candidate joined after transfer and subsequent semesters under the autonomous status. The class will be awarded based on the academic performance of a student in the autonomous pattern.

e) Readmission from IARE-R16/R18 to IARE-UG.20 regulations

A student took admission in IARE-R18 Regulations, detained due to lack of required number of credits or percentage of attendance at the end of any semester is permitted to take re-admission at appropriate level under any regulations prevailing in the institute subject to the following rules and regulations.

1. Student shall pass all the courses in the earlier scheme of regulations (IARE - R18). However, in case of having backlog courses, they shall be cleared by appearing for supplementary examinations conducted under IARE - R18 regulations from time to time.
2. After rejoining, the student is required to study the courses as prescribed in the new regulations for the re-admitted program at that level and thereafter.
3. If the student has already passed any course(s) of readmitted program in the earlier regulation / semester of study, such courses are exempted in the new scheme to appear for the course(s).
4. The courses that are not done in the earlier regulations / semester as compared with readmitted program need to be cleared after readmission by appearing for the examinations conducted time to time under the new regulations.
5. In general, after transition, course composition and number of credits / semester shall be balanced between earlier and new regulations on case to case basis.
6. In case, the students who do not have option of acquiring required credits with the existing courses offered as per the new curriculum, credit balance can be achieved by clearing the additional courses offered by the respective departments (approved in Academic Council meeting). The additional courses that are offered can be of theory or laboratory courses and shall be offered during semester.
7. Students re-joined in III semester shall be treated on par with "Lateral Entry" students for credits and graduation requirements. However, the student shall clear all the courses in B.Tech I Semester and B.Tech II Semester as per IARE-R18 regulations.

31. REVISION OF REGULATIONS AND CURRICULUM

The Institute from time to time may revise, amend or change the regulations, scheme of examinations and syllabi if found necessary and on approval by the Academic Council and the Governing Body and shall be binding on the students, faculty, staff, all authorities of the Institute and others concerned.

FREQUENTLY ASKED QUESTIONS AND ANSWERS ABOUT AUTONOMY

1. Who grants Autonomy? UGC, Govt., AICTE or University

In case of Colleges affiliated to a university and where statutes for grant of autonomy are ready, it is the respective University that finally grants autonomy but only after concurrence from the respective state Government as well as UGC. The State Government has its own powers to grant autonomy directly to Govt. and Govt. aided Colleges.

2 Shall IARE award its own Degrees?

No. Degree will be awarded by Jawaharlal Nehru Technological University, Hyderabad with a mention of the name IARE on the Degree Certificate.

3 What is the difference between a Deemed University and an Autonomy College?

A Deemed University is fully autonomous to the extent of awarding its own Degree. A Deemed University is usually a Non-Affiliating version of a University and has similar responsibilities like any University. An Autonomous College enjoys Academic Autonomy alone. The University to which an autonomous college is affiliated will have checks on the performance of the autonomous college.

4 How will the Foreign Universities or other stake – holders know that we are an Autonomous College?

Autonomous status, once declared, shall be accepted by all the stake holders. The Govt. of Telangana mentions autonomous status during the First Year admission procedure. Foreign Universities and Indian Industries will know our status through our website.

5 What is the change of Status for Students and Teachers if we become Autonomous?

An autonomous college carries a prestigious image. Autonomy is actually earned out of our continued past efforts on academic performances, our capability of self- governance and the kind of quality education we offer.

6 Who will check whether the academic standard is maintained / improved after Autonomy? How will it be checked?

There is a built in mechanism in the autonomous working for this purpose. An Internal Committee called Academic Program Evaluation Committee, which will keep a watch on the academics and keep its reports and recommendations every year. In addition the highest academic council also supervises the academic matters. The standards of our question papers, the regularity of academic calendar, attendance of students, speed and transparency of result declaration and such other parameters are involved in this process.

7 Will the students of IARE as an Autonomous College qualify for University Medals and Prizes for academic excellence?

No. IARE has instituted its own awards, medals, etc. for the academic performance of the students. However for all other events like sports, cultural on co-curricular organized by the University the students shall qualify.

8 Can IARE have its own Convocation?

No. Since the University awards the Degree the Convocation will be that of the University, but there will be Graduation Day at IARE.

9 Can IARE give a provisional degree certificate?

Since the examinations are conducted by IARE and the results are also declared by IARE, the college sends a list of successful candidates with their final Grades and Grade Point Averages including CGPA to the University. Therefore with the prior permission of the University the college will be entitled to give the provisional certificate.

10 Will Academic Autonomy make a positive impact on the Placements or Employability?

Certainly. The number of students qualifying for placement interviews is expected to improve, due to rigorous and repetitive classroom teaching and continuous assessment. Also the autonomous status is more responsive to the needs of the industry. As a result therefore, there will be a lot of scope for industry oriented skill development built-in into the system. The graduates from an autonomous college will therefore represent better employability.

11 What is the proportion of Internal and External Assessment as an Autonomous College?

Presently, it is 60% external and 40% internal. As the autonomy matures the internal assessment component shall be increased at the cost of external assessment.

12 Is it possible to have complete Internal Assessment for Theory or Practicals?

Yes indeed. We define our own system. We have the freedom to keep the proportion of external and internal assessment component to choose.

13 Why Credit based Grade System?

The credit based grade system is an accepted standard of academic performance the world over in all Universities. The acceptability of our graduates in the world market shall improve.

14 What exactly is a Credit based Grade System?

The credit based grade system defines a much better statistical way of judging the academic performance. One Lecture Hour per week of Teaching Learning process is assigned One Credit. One hour of laboratory work is assigned half credit. Letter Grades like A, B,C,D, etc. are assigned for a Range of Marks. (e.g. 91% and above is A+, 80 to 90 % could be A etc.) in Absolute Grading System while grades are awarded by statistical analysis in relative grading system. We thus dispense with sharp numerical boundaries. Secondly, the grades are associated with defined Grade Points in the scale of 1 to 10. Weighted Average of Grade Points is also defined Grade Points are weighted by Credits and averaged over total credits in a Semester. This process is repeated for all Semesters and a CGPA defines the Final Academic Performance

15 What are the norms for the number of Credits per Semester and total number of Credits for UG/PG program?

These norms are usually defined by UGC or AICTE. Usually around 25 Credits per semester is the accepted norm.

16 What is a Semester Grade Point Average (SGPA)?

The performance of a student in a semester is indicated by a number called SGPA. The SGPA is the weighted average of the grade points obtained in all the courses registered by the student during the semester.

$$SGPA = \sum_{i=1}^{n} (C_i G_i) / \sum_{i=1}^{n} C_i$$

Where, C_i is the number of credits of the i^{th} course and G_i is the grade point scored by the student in the i^{th} course and i represent the number of courses in which a student registered in the concerned semester. SGPA is rounded to two decimal places.

17 What is a Cumulative Grade Point Average (CGPA)?

An up-to-date assessment of overall performance of a student from the time of his first registration is obtained by calculating a number called CGPA, which is weighted average of the grade points obtained in all the courses registered by the students since he entered the Institute.

$$CGPA = \sum_{j=1}^{m} (C_j S_j) / \sum_{j=1}^{m} C_j$$

Where, S_j is the SGPA of the j^{th} semester and C_j is the total number of credits upto the semester and m represent the number of semesters completed in which a student registered upto the semester. CGPA is rounded to two decimal places.

18 Is there any Software available for calculating Grade point averages and converting the same into Grades?

Yes, The institute has its own MIS software for calculation of SGPA, CGPA, etc.

19 Will the teacher be required to do the job of calculating SGPAs etc. and convert the same into Grades?

No. The teacher has to give marks obtained out of whatever maximum marks as it is. Rest is all done by the computer.

20 Will there be any Revaluation or Re-Examination System?

No. There will double valuation of answer scripts. There will be a makeup Examination after a reasonable preparation time after the End Semester Examination for specific cases mentioned in the Rules and Regulations. In addition to this, there shall be a 'summer term' (compressed term) followed by the End Semester Exam, to save the precious time of students.

21 How fast Syllabi can be and should be changed?

Autonomy allows us the freedom to change the syllabi as often as we need.

22 Will the Degree be awarded on the basis of only final year performance?

No. The CGPA will reflect the average performance of all the semester taken together.

23 What are Statutory Academic Bodies?

Governing Body, Academic Council, Examination Committee and Board of Studies are the different statutory bodies. The participation of external members in everybody is compulsory. The institute has nominated professors from IIT, NIT, University (the officers of the rank of Pro-vice Chancellor, Deans and Controller of Examinations) and also the reputed industrialist and industry experts on these bodies.

24 Who takes Decisions on Academic matters?

The Governing Body of institute is the top academic body and is responsible for all the academic decisions. Many decisions are also taken at the lower level like Boards of Studies. Decisions taken at the Boared of Studies level are to be ratified at the Academic Council and Governing Body.

25 What is the role of Examination committee?

The Examinations Committee is responsible for the smooth conduct of internal, End Semester and make up Examinations. All matters involving the conduct of examinations spot valuations, tabulations preparation of Grade Sheet etc fall within the duties of the Examination Committee.

26 Is there any mechanism for Grievance Redressal?

The institute has grievance redressal committee, headed by Dean - Student affairs and Dean - IQAC.

27 How many attempts are permitted for obtaining a Degree?

All such matters are defined in Rules & Regulation

28 Who declares the result?

The result declaration process is also defined. After tabulation work wherein the SGPA, CGPA and final Grades are ready, the entire result is reviewed by the Moderation Committee. Any unusual deviations or gross level discrepancies are deliberated and removed. The entire result is discussed in the Examinations and Result Committee for its approval. The result is then declared on the institute notice boards as well put on the web site and Students Corner. It is eventually sent to the University.

29 Who will keep the Student Academic Records, University or IARE?

It is the responsibility of the Dean, Academics of the Autonomous College to keep and preserve all the records.

30 **What is our relationship with the JNT University?**
We remain an affiliated college of the JNT University. The University has the right to nominate its members on the academic bodies of the college.

31 **Shall we require University approval if we want to start any New Courses?**
Yes, It is expected that approvals or such other matters from an autonomous college will receive priority.

32 **Shall we get autonomy for PG and Doctoral Programs also?**
Yes, presently our PG programs also enjoying autonomous status.

MALPRACTICE RULES

DISCIPLINARY ACTION FOR / IMPROPER CONDUCT IN EXAMINATIONS

S. No	Nature of Malpractices/Improper conduct	Punishment
	If the candidate:	
1. (a)	Possesses or keeps accessible in examination hall, any paper, note book, programmable calculator, cell phone, pager, palm computer or any other form of material concerned with or related to the subject of the examination (theory or practical) in which he is appearing but has not made use of (material shall include any marks on the body of the candidate which can be used as an aid in the subject of the examination)	Expulsion from the examination hall and cancellation of the performance in that subject only.
(b)	Gives assistance or guidance or receives it from any other candidate orally or by any other body language methods or communicates through cell phones with any candidate or persons in or outside the exam hall in respect of any matter.	Expulsion from the examination hall and cancellation of the performance in that subject only of all the candidates involved. In case of an outsider, he will be handed over to the police and a case is registered against him.
2.	Has copied in the examination hall from any paper, book, programmable calculators, palm computers or any other form of material relevant to the subject of the examination (theory or practical) in which the candidate is appearing.	Expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted to appear for the remaining examinations of the subjects of that Semester/year. The Hall Ticket of the candidate is to be cancelled and sent to the Controller of Examinations.
3.	Impersonates any other candidate in connection with the examination.	The candidate who has impersonated shall be expelled from examination hall. The candidate is also debarred and forfeits the seat. The performance of the original candidate, who has been impersonated, shall be cancelled in all the subjects of the examination (including practicals and project work) already appeared and shall not be allowed to appear for examinations of the remaining subjects of that semester/year. The candidate is also debarred for two consecutive semesters from class work and all semester end examinations. The continuation of the course by the candidate is subject to the academic regulations in connection with forfeiture of seat. If the imposter is an outsider, he will be handed over to the police and a case is registered against him.
4.	Smuggles in the Answer book or additional sheet or takes out or arranges to send out the question paper during the examination or answer book or additional sheet, during or after the examination.	Expulsion from the examination hall and cancellation of performance in that subject and all the other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the

		remaining examinations of the subjects of that semester/year. The candidate is also debarred for two consecutive semesters from class work and all semester end examinations. The continuation of the course by the candidate is subject to the academic regulations in connection with forfeiture of seat.
5.	Uses objectionable, abusive or offensive language in the answer paper or in letters to the examiners or writes to the examiner requesting him to award pass marks.	Cancellation of the performance in that subject.
6.	Refuses to obey the orders of the Controller of Examinations /Additional Controller of Examinations/any officer on duty or misbehaves or creates disturbance of any kind in and around the examination hall or organizes a walk out or instigates others to walk out, or threatens the COE or any person on duty in or outside the examination hall of any injury to his person or to any of his relations whether by words, either spoken or written or by signs or by visible representation, assaults the COE or any person on duty in or outside the examination hall or any of his relations, or indulges in any other act of misconduct or mischief which result in damage to or destruction of property in the examination hall or any part of the Institute premises or engages in any other act which in the opinion of the officer on duty amounts to use of unfair means or misconduct or has the tendency to disrupt the orderly conduct of the examination.	In case of students of the college, they shall be expelled from examination halls and cancellation of their performance in that subject and all other subjects the candidate(s) has (have) already appeared and shall not be permitted to appear for the remaining examinations of the subjects of that semester/year. The candidates also are debarred and forfeit their seats. In case of outsiders, they will be handed over to the police and a police case is registered against them.
7.	Leaves the exam hall taking away answer script or intentionally tears off the script or any part thereof inside or outside the examination hall.	Expulsion from the examination hall and cancellation of performance in that subject and all the other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year. The candidate is also debarred for two consecutive semesters from class work and all semester end examinations. The continuation of the course by the candidate is subject to the academic regulations in connection with forfeiture of seat.
8.	Possess any lethal weapon or firearm in the examination hall.	Expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year. The candidate is also debarred and forfeits the seat.
9.	If student of the college, who is not a candidate for the particular examination or any person not connected with the college indulges in any	Student of the colleges expulsion from the examination hall and cancellation of the performance in that subject and all other subjects

	malpractice or improper conduct mentioned in clause 6 to 8.	the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year. The candidate is also debarred and forfeits the seat. Person(s) who do not belong to the College will be handed over to police and, a police case will be registered against them.
10.	Comes in a drunken condition to the examination hall.	Expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year.
11.	Copying detected on the basis of internal evidence, such as, during valuation or during special scrutiny.	Cancellation of the performance in that subject and all other subjects the candidate has appeared including practical examinations and project work of that semester/year examinations.
12.	If any malpractice is detected which is not covered in the above clauses 1 to 11 shall be reported to the University for further action to award suitable punishment.	

FAILURE TO READ AND UNDERSTAND THE REGULATIONS IS NOT AN EXCUSE

INSTITUTE OF AERONAUTICAL ENGINEERING

(Autonomous)

Dundigal, Hyderabad – 500043

COURSE CATALOG

(ELECTRICAL AND ELECTRONICS ENGINEERING)

I SEMESTER

Course Code	Course Name	Subject Area	Category	Periods per week			Credits	Scheme of Examination Max. Marks		
				L	T	P		CIA	SEE	Total
THEORY										
AHSC01	English	HSMC	Foundation	3	0	0	3	30	70	100
AHSC02	Linear Algebra and Calculus	BSC	Foundation	3	1	0	4	30	70	100
AHSC03	Engineering Physics	BSC	Foundation	3	0	0	3	30	70	100
ACSC01	Python Programming	ESC	Foundation	3	0	0	3	30	70	100
PRACTICAL										
AHSC04	English Language and Communication Skills Laboratory	HSMC	Foundation	0	0	2	1	30	70	100
AHSC05	Physics Laboratory	BSC	Foundation	0	0	3	1.5	30	70	100
ACSC02	Python Programming Laboratory	ESC	Foundation	0	0	3	1.5	30	70	100
TOTAL				**12**	**01**	**08**	**17**	**210**	**490**	**700**

II SEMESTER

Course Code	Course Name	Subject Area	Category	Periods per week			Credits	Scheme of Examination Max. Marks		
				L	T	P		CIA	SEE	Total
THEORY										
AHSC06	Chemistry	BSC	Foundation	2	0	0	2	30	70	100
AHSC07	Mathematical Transform Techniques	BSC	Foundation	3	1	0	4	30	70	100
AEEC02	Electrical Circuits	ESC	Foundation	3	0	0	3	30	70	100
ACSC04	Programming for Problem Solving using C	ESC	Foundation	3	0	0	3	30	70	100
ACSC06	Experiential Engineering Education (ExEEd) – Academic Sucess	ESC	Foundation	2	0	0	1	30	70	100
PRACTICAL										
AEEC03	Electrical Circuits Laboratory	ESC	Foundation	0	0	3	1.5	30	70	100
ACSC05	Programming for Problem Solving using C Laboratory	ESC	Foundation	0	0	3	1.5	30	70	100
AMEC04	Engineering Workshop Practice	ESC	Foundation	0	0	2	1	30	70	100
TOTAL				**13**	**01**	**08**	**17**	**240**	**560**	**800**

III SEMESTER

Course Code	Course Name	Subject Area	Category	Periods per week			Credits	Scheme of Examination Max. Marks		
				L	T	P		CIA	SEE	Total
THEORY										
AEEC05	Network Analysis	PCC	Core	3	0	0	3	30	70	100
AEEC06	Electromagnetic Fields	PCC	Core	3	1	0	4	30	70	100
AECC07	Analog Electronics	PCC	Core	3	0	0	3	30	70	100
AEEC07	DC Machines and Transformers	PCC	Core	3	0	0	3	30	70	100
ACSC08	Data Structures	PCC	Core	3	0	0	3	30	70	100
ACSC09	ExEEd - Prototype / Design Building	ESC	Foundation	2	0	0	1	30	70	100
PRACTICALS										
AEEC08	Network Analysis and Scientific Computing Laboratory	PCC	Core	0	1	2	2	30	70	100
AEEC09	DC Machines Laboratory	PCC	Core	0	0	3	1.5	30	70	100
ACSC10	Data Structures Laboratory	PCC	Core	0	0	3	1.5	30	70	100
MANDATORY / VALUE ADDED COURSES										
AHSC10	Essence of Indian Traditional Knowledge	MC	MC	Ref: 8.4 Academic Regulations, UG.20						
TOTAL				**17**	**02**	**08**	**22**	**270**	**630**	**900**

IV SEMESTER

Course Code	Course Name	Subject Area	Category	Periods per week			Credits	Scheme of Examination Max. Marks		
				L	T	P		CIA	SEE	Total
THEORY										
AHSC11	Complex Analysis and Probability Distributions	BSC	Foundation	3	1	0	4	30	70	100
AEEC10	Electrical Power Generation Systems	PCC	Core	3	1	0	4	30	70	100
AEEC11	AC Machines	PCC	Core	3	0	0	3	30	70	100
AEEC12	Control Systems	PCC	Core	3	0	0	3	30	70	100
AECC16	Digital Electronics	PCC	Core	3	0	0	3	30	70	100
ACSC14	ExEEd - Fabrication / Model Development	ESC	Foundation	2	0	0	1	30	70	100
PRACTICALS										
AEEC13	AC Machines Laboratory	PCC	Core	0	0	3	1.5	30	70	100
AEEC14	Control Systems Laboratory	PCC	Core	0	0	3	1.5	30	70	100
AECC17	Analog and Digital Electronics Laboratory	PCC	Core	0	0	2	1	30	70	100
MANDATORY / VALUE ADDED COURSES										
ACSC18	Fundamentals of Database Systems	Skill	VAC	Ref: 8.6, Academic Regulations-UG.20						
	TOTAL			**17**	**02**	**08**	**22**	**270**	**630**	**900**

V SEMESTER

Course Code	Course Name	Subject Area	Category	Periods per week			Credits	Scheme of Examination Max. Marks		
				L	T	P		CIA	SEE	Total
THEORY										
AHSC13	Business Economics and Financial Analysis	HSMC	Foundation	3	0	0	3	30	70	100
AEEC15	Electrical Power Transmission Systems	PCC	Core	3	1	0	4	30	70	100
AEEC16	Power Electronics	PCC	Core	3	1	0	4	30	70	100
AECC19	Microprocessors and Microcontrollers	PCC	Core	3	1	0	4	30	70	100
	Professional Elective - I	PEC	Elective	3	0	0	3	30	70	100
ACSC20	ExEEd - Project Based Learning	ESC	Foundation	2	0	0	1	30	70	100
PRACTICALS										
AEEC21	Power Electronics Laboratory	PCC	Core	0	0	3	1.5	30	70	100
AECC31	Microprocessors and Microcontrollers Laboratory	PCC	Core	0	0	3	1.5	30	70	100
MANDATORY / VALUE ADDED COURSES										
ACSC23	Object Oriented Programming Development and Languages	VAC	Skill	Ref: 8.6, Academic Regulations-UG.20						
TOTAL				**17**	**03**	**06**	**22**	**240**	**560**	**800**

VI SEMESTER

Course Code	Course Name	Subject Area	Category	Periods per week			Credits	Scheme of Examination Max. Marks		
				L	T	P		CIA	SEE	Total
THEORY										
AEEC22	Power System Analysis	PCC	Core	3	1	0	4	30	70	100
AEEC23	Electric Drives and Static Control	PCC	Core	3	1	0	4	30	70	100
AEEC24	Electrical Measurements and Instrumentation	PCC	Core	3	0	0	3	30	70	100
	Professional Elective - II	PEC	Elective	3	0	0	3	30	70	100
	Open Elective - I	OEC	Elective	3	0	0	3	30	70	100
ACSC27	ExEEd - Research Based Learning	ESC	Foundation	2	0	0	1	30	70	100
PRACTICALS										
AEEC32	PLC and Industrial Automation Laboratory	PCC	Core	1	0	3	2.5	30	70	100
AEEC33	Electrical Measurements and Instrumentation Laboratory	PCC	Core	0	0	3	1.5	30	70	100
MANDATORY / VALUE ADDED COURSES										
ACSC29	Design of Algorithms	VAC	Skill	Ref: 8.6, Academic Regulations-UG.20						
TOTAL				**18**	**02**	**06**	**22**	**240**	**560**	**800**

VII SEMESTER

Course Code	Course Name	Subject Area	Category	Periods per week			Credits	Scheme of Examination Max. Marks		
				L	T	P		CIA	SEE	Total
THEORY										
AEEC34	Power System Protection	PCC	Core	3	1	0	4	30	70	100
AEEC35	Power System Operation and Control	PCC	Core	3	0	0	3	30	70	100
	Professional Elective - III	PEC	Elective	3	0	0	3	30	70	100
	Professional Elective - IV	PEC	Elective	3	0	0	3	30	70	100
	Open Elective - II	OEC	Elective	3	0	0	3	30	70	100
PRACTICALS										
AEEC44	Electrical Power Systems Laboratory	PCC	Core	0	0	3	1.5	30	70	100
AEEC45	Power System Simulation Laboratory	PCC	Core	0	0	3	1.5	30	70	100
AEEC46	Project Work (Phase - I)	PROJ	Project	0	0	4	2	30	70	100
TOTAL				**15**	**01**	**10**	**21**	**240**	**560**	**800**

VIII SEMESTER

Course Code	Course Name	Subject Area	Category	Periods per week			Credits	Scheme of Examination Max. Marks		
				L	T	P		CIA	SEE	Total
THEORY										
	Professional Elective - V	PEC	Elective	3	0	0	3	30	70	100
	Professional Elective - VI	PEC	Elective	3	0	0	3	30	70	100
	Open Elective - III	OEC	Elective	3	0	0	3	30	70	100
PRACTICALS										
AEEC55	Project Work (Phase - II)	PCC	Project	0	0	16	8	30	70	100
TOTAL				**09**	**00**	**16**	**17**	**120**	**280**	**400**

PROFESSIONAL ELECTIVES

Course code	PE - I Electrical Engineering	Course code	PE - II Systems and Controls	Course code	PE - III Stability Studies
AEEC17	Electrical Machine Design	AEEC25	Digital Control Systems	AEEC36	Power System Stability
AEEC18	Computational Electromagnetics	AEEC26	Principles of Signals and Systems	AEEC37	Power System Dynamics and Control
AEEC19	Special Electrical Machines	AEEC27	Embedded Systems and IoT	AEEC38	Control Systems Design
AEEC20	Electrical Energy Conservation and Auditing	AEEC28	Linear System Analysis	AEEC39	Digital Signal Processing and Applications

Course code	PE - IV Generation and Transmission	Course code	PE - V Advanced Power Systems	Course code	PE - VI Power Systems and Control
AEEC40	HVDC Transmission	AEEC47	High Voltage Engineering	AEEC51	Utilization of Electric Power
AEEC41	EHVAC Transmission	AEEC48	Energy Storage Systems	AEEC52	Industrial Electrical Systems
AEEC42	Power Electronics in Renewable Energy Systems	AEEC49	Power Quality and FACTS	AEEC53	Renewable Energy and Smart Grid
AEEC43	Wind and Solar Energy Systems	AEEC50	Switch Mode Power Supplies	AEEC54	Electrical and Hybrid Vehicles

OPEN ELECTIVES COURSES

OPEN ELECTIVES – I

Course Code	Course Title
ACSC24	Computer Architecture
ACSC25	Advanced Data Structures
ACSC26	Artificial Intelligence
AITC19	Cyber Crime and Computer Forensics
AITC20	Ethical Hacking
AITC21	Mobile Computing

OPEN ELECTIVE - II

Course Code	Course Title
AHSC15	Soft Skills and Interpersonal Communication
AHSC16	Cyber Law and Ethics
AHSC17	Economic Policies in India
AHSC18	Global Warming and Climate Change
AHSC19	Intellectual Property Rights
AHSC20	Entrepreneurship

OPEN ELECTIVE – III

Course Code	Course Title
AAEC30	Flight Control Theory
AAEC31	Airframe Structural Design
AMEC34	Industrial Management
AMEC35	Elements of Mechanical Engineering
ACEC30	Modern Construction Materials
ACEC31	Disaster Management

VALUE ADDED COURSES / MANDATORY COURSES

Course Code	Course Title
AHSC10	Essence of Indian Traditional Knowledge (MC)
ACSC18	Fundamentals of Database Systems (VAC)
ACSC23	Object Oriented Programming Development and Languages (VAC)
ACSC29	Design of Algorithms (VAC)

SYLLABUS
(I - VIII SEMESTERS)

ENGLISH

I Semester: AE / ECE / EEE / ME / CE II Semester : CSE / CSE (AI & ML) / CSE (DS) / CSE (CS) / CSIT / IT								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
AHSC01	**Foundation**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		2	-	-	2	30	70	100
Contact Classes: 45	**Tutorial Classes: Nil**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite: **Standard applicability of vocabulary and grammer**								

I. COURSE OVERVIEW:

The sole aim of the course is to enhance the communication skills of upcoming engineering graduates to meet the requirements and challenges in a competitive global world. This course is designed to provide a well-rounded introduction to English language learning. Moreover, the course pays special attention to the typical problems and challenges confronted by the Indian learners of English like mispronunciation, spellings, and structures of English due to their mother tongue influence. This course includes General Introduction to Listening Skills, Speaking Skills, Vocabulary and Grammar, Reading Skills, and Writing Skills.

II. COURSE OBJECTIVES:

The Students will try to learn:

I. The theoretical and fundamental inputs to communicate intelligibly in English through standard Pronunciation.
II. The four language skills i.e., Listening, Speaking, Reading and Writing effectively and their application in real-life situations.
III. The Writing strategies of English using correct spelling, grammar, punctuation and appropriate vocabulary.
IV. Different mechanics of writing styles forms of writing emails, reports, formal and informal letters.

III. COURSE SYLLABUS:

MODULE-I: GENERAL INTRODUCTION AND LISTENING SKILLS (09)

Introduction to communication skills; Communication process; Elements of communication; Soft skills vs hard skills; Listening skills; Significance; Stages of listening; Barriers to listening and effectiveness of listening; Listening comprehension.

MODULE –II: SPEAKING SKILLS (09)

Significance; Essentials; Barriers and effectiveness of speaking; Verbal and non-verbal communication; Generating talks based on visual prompts; Public speaking; Exposure to structured talks; Addressing a small group or a large formal gathering; Oral presentation.

MODULE –III: VOCABULARY & GRAMMAR (09)

Vocabulary: The concept of Word Formation; Root words from foreign languages and their use in English; Acquaintance with prefixes and suffixes from foreign languages in English to form derivatives; Idioms and phrases; One-word substitutes.

Grammar: Sentence structure; Uses of phrases and clauses; Punctuation; Subject verb agreement; Modifiers; Articles; Prepositions.

MODULE –IV: READING SKILLS (09)

Significance; Techniques of reading; Skimming-Reading for the gist of a text; Scanning - Reading for specific information; Intensive; Extensive reading; Reading comprehension; Reading for information transfer; Text to diagram; Diagram to text.

MODULE –V: WRITING SKILLS (09)

Significance; Effectiveness of writing; Organizing principles of Paragraphs in documents; Techniques for writing precisely; Letter writing; Formal and Informal letter writing; E-mail writing, Report Writing.

IV. TEXT BOOKS:

1. Handbook of English for Communication (Prepared by Faculty of English, IARE).

V. REFERENCE BOOKS:

1. Sanjay Kumar and Pushp Lata. “Communications Skills”. Oxford University Press. 2011.
2. Michael Swan. “Practical English Usage”, Oxford University Press, 1995.
3. F.T. Wood. “Remedial English Grammar”. Macmillan. 2007.
4. William Zinsser. “On Writing Well”. Harper Resource Book, 2001.
5. Raymond Murphy, “Essential English Grammar with Answers”, Cambridge University Press 2nd Edition, 2011.

VI. WEB REFERENCES:

1. www.edufind.com
2. www.myenglishpages.com
3. http:grammar.ccc.comment.edu
4. http:owl.english.prudue.edu

VII. E-TEXT BOOKS:

1. http://bookboon.com/en/communication-ebooks-zip
2. http://www.bloomsbury-international.com/images/ezone/ebook/writing-skills-pdf.pdf
3. https://americanenglish.state.gov/files/ae/resource_files/developing_writing.pdf
4. http://learningenglishvocabularygrammar.com/files/idiomsandphraseswithmeaningsandexamplespdf
5. http://www.robinwood.com/Democracy/GeneralEssays/CriticalThinking.pdf

LINEAR ALGEBRA AND CALCULUS

<table>
<tr><td colspan="10">I Semester: Common for All Branches</td></tr>
<tr><th>Course Code</th><th>Category</th><th colspan="3">Hours / Week</th><th>Credits</th><th colspan="3">Maximum Marks</th></tr>
<tr><td rowspan="2">AHSC02</td><td rowspan="2">Foundation</td><td>L</td><td>T</td><td>P</td><td>C</td><td>CIA</td><td>SEE</td><td>Total</td></tr>
<tr><td>3</td><td>1</td><td>-</td><td>4</td><td>30</td><td>70</td><td>100</td></tr>
<tr><td>Contact Classes: 45</td><td>Tutorial Classes: 15</td><td colspan="4">Practical Classes: Nil</td><td colspan="3">Total Classes: 60</td></tr>
<tr><td colspan="10">Prerequisite: Basic principles of algebra and calculus</td></tr>
</table>

I. COURSE OVERVIEW:

Linear algebra is a sub-field of mathematics concerned with vectors, matrices, and linear transforms. Calculus is the branch of mathematics which majorly deals with derivatives and integrals. Linear algebra is a key foundation to the field of machine learning. Matrices are used in computer animations, color image processing. Eigenvalues are used by engineers to discover new and better designs for the future. Differential equations have wide applications in various engineering and science disciplines as the laws of physics are generally written down as differential equations.The Fourier series has many applications in electrical engineering, image processing etc.The course includes types of Matrices, Rank, methods of finding rank, eigen values and eigen vectors, maxima and minima of functions of several variables, solutions of higher order ordinary differential equations and Fourier series.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The principles of Eigen value analysis and linear transformations, Matrix rank finding methods
II. The calculus of functions of several variables and the concept of maxima-minima for a three-dimensional surface.
III. The analytical methods for solving higher order differential equations with constant coefficients.
IV. Fourier series expansions in standard intervals as well as arbitrary intervals.

III. COURSE SYLLABUS:

MODULE-I: THEORY OF MATRICES (09)

Real matrices: Symmetric, skew-symmetric and orthogonal matrices; Complex matrices: Hermitian, Skew- Hermitian and unitary matrices; Elementary row and column transformations, finding rank of a matrix by reducing to Echelon form and Normal form; Finding the inverse of a matrix using Gauss-Jordan method;

MODULE –II: LINEAR TRANSFORMATIONS (09)

Cayley-Hamilton theorem: Statement, verification, finding inverse and powers of a matrix; Linear dependence and independence of vectors; Linear transformation; Eigen values and Eigen vectors of a matrix; Diagonalization of matrix by linear transformation.

MODULE –III: FUNCTIONS OF SINGLE AND SEVERAL VARIABLES (09)

Mean value theorems: Rolle's theorem, Lagrange's theorem, Cauchy's theorem-without proof.

Functions of several variables: Partial differentiation, Jacobian, functional dependence, maxima and minima of functions with two variables and three variables. Method of Lagrange multipliers.

MODULE –IV: HIGHER ORDER LINEAR DIFFERENTIAL EQUATIONS (09)

Linear differential equations of second and higher order with constant coefficients.

Non-homogeneous term of the type $f(x)=e^{ax}, \sin ax, \cos ax$ and $f(x)=x^n, e^{ax}v(x)$, Method of variation of parameters.

MODULE –V: FOURIER SERIES (09)

Fourier expansion of periodic function in a given interval of length 2π; Fourier series of even and odd functions; Fourier series in an arbitrary interval, Half- range Fourier sine and cosine expansions.

IV. TEXT BOOKS

1. B.S. Grewal, Higher Engineering Mathematics, Khanna Publishers, 36^{th} Edition, 2010.
2. N.P. Bali and Manish Goyal, A text book of Engineering Mathematics, Laxmi Publications, Reprint, 2008.
3. Ramana B.V., Higher Engineering Mathematics, Tata McGraw Hill New Delhi, 11^{th} Reprint 2010.

V. REFERENCE BOOKS:

1. Erwin Kreyszig, Advanced Engineering Mathematics, 9^{th} Edition, John Wiley & Sons, 2006.
2. Veerarajan T., Engineering Mathematics for first year, Tata McGraw-Hill, New Delhi, 2008.
3. D. Poole, Linear Algebra: A Modern Introduction, 2^{nd} Edition, Brooks/Cole, 2005.

VI. WEB REFERENCES:

1. http://www.efunda.com/math/math_home/math.cfm
2. http://www.ocw.mit.edu/resources/#Mathematics
3. http://www.sosmath.com
4. http://www.mathworld.wolfram.com

VII. E-TEXT BOOKS:

1. http://www.e-booksdirectory.com/details.php?ebook=10166
2. http://www.e-booksdirectory.com/details.php?ebook=7400re

ENGINEERING PHYSICS

I Semester: AE / ME / CE / EEE / ECE

Course Code	Category	Hours / Week			Credits	Maximum Marks		
AHSC03	Foundation	L	T	P	C	CIA	SEE	Total
		3	-	-	3	30	70	100
Contact Classes: 45	Tutorial Classes: Nil	Practical Classes: Nil				Total Classes: 45		

Prerequisite: Basic principles of waves

I. COURSE OVERVIEW:

This course develops abstract and critical reasoning by studying mathematical and logical proofs and assumptions as applied in basic physics and to make connections between physics and other branches of sciences and technology. The topics covered include waves, non-dispersive transverse and longitudinal waves, light and optics, wave optics, lasers, introduction to quantum mechanics, solution of wave equation and introduction to solids and semiconductors. The course helps students to gain knowledge of basic principles and appreciate the diverse applications in technological fields in respective branches.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The basic formulations in wave mechanics for the evolution of energy levels and quantization of energies for a particle in a potential box with the help of mathematical description

II. The fundamental properties of semiconductors including the band gap, charge carrier concentration, doping and charge carrier transport mechanisms

III. The simple optical setups and experimental approaches of Light and Laser using its interaction with matter

IV. The basic studies between different harmonic oscillators and different waves for using those relationships on practical problems.

III. COURSE OBJECTIVES:

MODULE-I: QUANTUM MECHANICS (09)

Introduction to quantum physics, de-broglie's hypothesis, Wave-particle duality, Davisson and Germer experiment, Time-independent Schrodinger equation for wave function, Physical significance of the wave function, Schrodinger equation for one dimensional problems–particle in a box.

MODULE –II: INTRODUCTION TO SOLIDS AND SEMICONDUCTORS (09)

Introduction to classical free electron theory and quantum theory, Bloch's theorem for particles in a periodic potential (Qualitative treatment), Kronig-Penney model (Qualitative treatment), classification: metals, semiconductors, and insulators. Intrinsic and extrinsic semiconductors, Carrier concentration, Dependence of Fermi level on carrier-concentration and temperature, Hall effect.

MODULE –III: LASERS AND FIBER OPTICS (09)

Characteristics of lasers, Spontaneous and stimulated emission of radiation, Metastable state, Population inversion, Lasing action, Ruby laser, He-Ne laser and applications of lasers.

Principle and construction of an optical fiber, Acceptance angle, Numerical aperture, Types of optical fibers (Single mode, multimode, step index, graded index), Optical fiber communication system with block diagram and Applications of optical fibers.

MODULE –IV: LIGHT AND OPTICS (09)

Principle of superposition of waves, Young's double slit experiment, Fringe width, Newton's rings.

Fraunhofer diffraction from a single slit, double slit (extension to N slits) and diffraction grating experiment.

MODULE –V: HARMONIC OSCILLATIONS AND WAVES IN ONE DIMENSION (09)

Simple harmonic oscillator, Damped harmonic oscillator, Forced harmonic oscillator. Transverse waves and Longitudinal wave equation, Reflection and transmission of waves at a boundary, Harmonic waves.

IV. TEXT BOOKS:

1. G. Main, “Vibrations and Waves in Physics”, Cambridge University Press, 1993.
2. R. K. Gaur, S. L. Gupta, “Engineering Physics”, Dhanpat Rai Publications, 8^{th} Edition, 2001.
3. Dr. K. Vijaya Kumar, Dr. S. Chandralingam, “Modern Engineering Physics”, Chand & Co. New Delhi, 1^{st} Edition, 2010.

V. REFERENCE BOOKS:

1. H.J. Pain, “The Physics of Vibrations and Waves”, Wiley, 2006.
2. Ghatak, “Optics”, McGraw Hill Education, 2012.
3. O. Svelto, “Principles of Lasers”, Springer Science & Business Media, 2010.

VI. WEB REFERENCES:

1. http://link.springer.com/book
2. http://www.thphys.physics.ox.ac.uk
3. http://www.sciencedirect.com/science
4. http://www.e-booksdirectory.com

VII. E-TEXT BOOKS:

1. http://www.peaceone.net/basic/Feynman/
2. http://physicsdatabase.com/free-physics-books/
3. http://www.damtp.cam.ac.uk/user/tong/statphys/sp.pdf
4. www.freebookcentre.net/Physics/Solid-State-Physics-Books.html

PYTHON PROGRAMMING

I Semester: Common for all branches									
Course Code	Category	Hours / Week			Credits	Maximum Marks		
ACSC01	Foundation	L	T	P	C	CIA	SEE	Total
		3	0	0	3	30	70	100
Contact Classes: 45	Tutorial Classes: Nil	Practical Classes: Nil				Total Classes: 45		
Prerequisites: There are no prerequisites to take this course.								

I. COURSE OVERVIEW:

This course introduces students to writing computer programs. This course presents the principles of structured programming using the Python language, one of the most increasingly preferred languages for programming today. Because of its ease of use, it is ideal as a first programming language and runs on both the PC and Macintosh platforms. However, the knowledge gained in the course can be applied later to other languages such as C and Java. The course uses iPython Notebook to afford a more interactive experience. Topics include fundamentals of computer programming in Python, object-oriented programming and graphical user interfaces.

II. COURSE OBJECTIVES:

The students will try to learn:

I. Acquire programming skills in core Python.
II. Acquire Object-oriented programming skills in Python.
III. Develop the skill of designing graphical-user interfaces (GUI) in Python.
IV. Develop the ability to write database applications in Python.
V. Acquire Python programming skills to move into specific branches - Internet of Things (IoT), Data Science, Machine Learning (ML), Artificial Intelligence (AI) etc.

III. SYLLABUS:

MODULE – I: INTRODUCTION TO PYTHON (09)

Introduction to Python: Features of Python, History and Future of Python, Working with Python – interactive and script mode, Identifiers and Keywords, Comments, Indentation and Multi-lining, Data types – built-in data types, Operators and Expressions, Console Input/Output, Formatted printing, Built-in Functions, Library Functions.

MODULE – II: DECISION CONTROL STATEMENTS (09)

Selection/Conditional Branching Statements: if, if-else, nested if, if-elif-else statement(s), Basic Loop Structures/ Iterative Statements – while and for loop, Nested loops, break and continue statement, pass Statement, else Statement used with loops.

MODULE – III: CONTAINER DATA TYPES (09)

Lists: Accessing List elements, List operations, List methods, List comprehension; Tuples: Accessing Tuple elements, Tuple operations, Tuple methods, Tuple comprehension, Conversion of List comprehension to Tuple, Iterators and Iterables, zip() function.

Sets: Accessing Set elements, Set operations, Set functions, Set comprehension; Dictionaries: Accessing Dictionary elements, Dictionary operations, Dictionary Functions, Nested Dictionary, Dictionary comprehension.

MODULE - IV STRINGS AND FUNCTIONS (09)

Strings: Accessing String elements, String properties, String operations.
Functions: Communicating with functions, Variable Scope and lifetime, return statement, Types of arguments, Lambda functions, Recursive functions.

MODULE - V CLASSES AND OBJECTS (09)

Classes and Objects – Defining Classes, Creating Objects, Data Abstraction and Hiding through Classes, Class Method and self Argument, Class variables and Object variables, __init()__ and __del__() method, Public and private data members, Built-in Class Attributes, Garbage Collection. OOPs Features: Abstraction, Encapsulation, Inheritance, and Polymorphism.

IV. TEXT BOOKS:

1. Reema Thareja, “Python Programming - Using Problem Solving Approach”, Oxford Press, 1st Edition, 2017.
2. Dusty Philips, “Python 3 Object Oriented Programming”, PACKT Publishing, 2nd Edition, 2015.

V. REFERENCE BOOKS:

1. Yashavant Kanetkar, Aditya Kanetkar, “Let Us Python”, BPB Publications, 2nd Edition, 2019.
2. Martin C. Brown, “Python: The Complete Reference”, Mc. Graw Hill, Indian Edition, 2018.
3. Michael H.Goldwasser, David Letscher, “Object Oriented Programming in Python”, Prentice Hall, 1st Edition, 2007.
4. Taneja Sheetal, Kumar Naveen, “Python Programming – A Modular Approach”, Pearson, 1st Edition, 2017.
5. R Nageswar Rao, “Core Python Programming”, Dreamtech Press, 2018.

VI. WEB REFERENCES:

1. https://realPython.com/Python3-object-oriented-programming/
2. https://Python.swaroopch.com/oop.html
3. https://Python-textbok.readthedocs.io/en/1.0/Object_Oriented_Programming.html
4. https://www.programiz.com/Python-programming/

ENGLISH LANGUAGE AND COMMUNICATION SKILLS LABORATORY

I SEMESTER: AE / ECE / EEE / ME / CE

II SEMESTER: CSE / CSE (AI & ML) / CSE (DS) / CSE (CS) / CSIT / IT

Course Code	Category	Hours / Week			Credits	Maximum Marks		
		L	T	P	C	CIA	SEE	Total
AHSC04	Foundation	-	-	2	1	30	70	100
Contact Classes: 45	Tutorial Classes: Nil	Practical Classes: 45				Total Classes: 45		

Prerequisite: There are no prerequisites to take this course.

I. COURSE OVERVIEW

The sole aim of the course is to enhance the communication skills of upcoming engineering graduates to meet the requirements and challenges in a competitive global world. This course includes General Introduction to Listening Skills, Speaking Skills, Vocabulary and Grammar, Reading Skills, and Writing Skills.

II. COURSE OBJECTIVES:

The students will try to:

I. Improve their ability to listen and comprehend a given text.
II. Upgrade the fluency and acquire a functional knowledge of English Language.
III. Enrich thought process by viewing a problem through multiple angles.

III. COURSE SYLLABUS:

Week-l: LISTENING SKILL

a. Listening to conversations and interviews of famous personalities in various fields; Listening practice related to the TV talk shows and news.
b. Listening for specific information; Listening for summarizing information – Testing..

Week-2: LISTENING SKILL

a. Listening to films of short duration and monologues for taking notes; Listening to answer multiple choice questions.
b. Listening to telephonic conversations; Listening to native Indian: Abdul Kalam, British: Helen Keller and American: Barrack Obama speakers to analyze intercultural differences – Testing.

Week-3: SPEAKING SKILL

a. Functions of English Language; Introduction to pronunciation; Vowels and Consonants
b. Tips on how to develop fluency, body language and communication; Introducing oneself: Talking about yourself, others, leave taking.

Week-4: SPEAKING SKILL

a. Sounds - Speaking exercises involving the use of Vowels and Consonant sounds in different contexts;Exercises on Homophones and Homographs
b. Just a minute (JAM) session.

Week-5: SPEAKING SKILL

a. Stress patterns.
b. Situational Conversations: common everyday situations; Acting as a compere and newsreader; Greetings for differentoccasionswithfeedbackpreferablythroughvideorecording.

Week-6: READING SKILL

a. Intonation.
b. Reading newspaper and magazine articles; Reading selective autobiographies for critical commentary.

Week-7: READING SKILL

a. Improving pronunciation through tongue twisters.
b. Reading advertisements, pamphlets; Reading comprehension exercises with critical and analytical questions based on context.

Week-8: WRITING SKILL

a. Listening to inspirational short stories and Writing messages
b. Writing leaflets, Notice; Writing tasks; Flashcards – Exercises

Week-9: WRITING SKILL

a. Write the review on a video clipping of short duration (5 to 10minutes).
b. Write a slogan related to the image; Write a short story of 6-10 lines based on the hints given.

Week-10: WRITING SKILL

a. Minimising Mother Tongue interference to improve fluency through watching educational videos.
b. Writing practices – précis writing; Essay writing

Week-11: THINKING SKILL

a. Correcting common errors in day to day conversations.

Practice in preparing thinking blocks to decode diagrammatical representations into English words,expressions, idioms, proverbs.

IV. TEXT BOOK:

1. "English Language and Communication Skills" Lab Manual - Prepared by the faculty of English, IARE.

V. REFERENCE BOOKS:

1. Meenakshi Raman, Sangeetha Sharma, "Technical Communication Principles and Practices", Oxford University Press, New Delhi, 3rd Edition, 2015.
2. Rhirdion, Daniel, "Technical Communication", Cengage Learning, New Delhi, 1st Edition, 2009.

PHYSICS LABORATORY

I Semester: AE / ME / CE / ECE / EEE II Semester: CSE / CSE (AI & ML) / CSE (DS) / CSE (CS) / CSIT / IT								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
AHSC05	Foundation	L	T	P	C	CIA	SEE	Total
		-	-	3	1.5	30	70	100
Contact Classes: Nil	Tutorial Classes: Nil	Practical Classes: 36			Total Classes: 36			
Pre-Requisites: Basic principles of Physics								

I. COURSE OVERVIEW:

This course is designed to lay a strong foundation in Engineering Physics that forms a basis to various branches of Engineering. It helps the students to perform experiments, to correlate theory with experimental data, analyse using graphical representations and present them as part of a clear, well-organized lab report. At the end of the course, students will be able to demonstrate a working knowledge of fundamentals of Physics and communicate their ideas effectively, both orally and in writing.

II. COURSE OBJECTIVES:

The students will try to learn:

1. Experimental skills in using optical instruments to determine physical constants.
2. The real time applications of electromagnetic theory.
3. The working principles of various electronic devices.

III. COURSE SYLLABUS:

Week-1: HALL EFFECT (LORENTZ FORCE)
Determination of charge carrier density.

Week-2: MELDE'E EXPERIMENT
Determination of frequency of a given tuning fork.

Week-3: STEWART GEE'S APPARATUS
Magnetic field along the axis of current carrying coil-Stewart and Gee‘s method.

Week-4: B-H CURVE WITH CRO
To determine the energy loss per unit volume of a given magnetic material per cycle by tracing the Hysteresis loop (B-H curve).

Week-5: ENERGY GAP OF A SEMICONDUCTOR DIODE
Determination of energy gap of a semiconductor diode.

Week-6: PHOTO DIODE
Studying V-I characteristics of photo diode.

Week-7: OPTICAL FIBER
Evaluation of numerical aperture of a given optical fiber.

Week-8: WAVE LENGTH OF LASER LIGHT
Determination of wavelength of a given laser light using diffraction grating.

Week-9: PLANCK'S CONSTANT
Determination of Planck‘s constant using LED.

Week-10: LIGHT EMITTING DIODE
Studying V-I characteristics of LED

Week-11: NEWTONS RINGS
Determination of radius of curvature of a given plano-convex lens.

Week-12: SINGLE SLIT DIFFRACTION

Determination of width of a given single slit.

IV. MANUALS:

1. C. L. Arora, “Practical Physics”, S. Chand & Co., New Delhi, 3rd Edition, 2012.
2. VijayKumar, Dr.T.Radhakrishna, “Practical Physics for Engineering Students”, SM Enterprises, 2nd Edition, 2014.

V. WEB REFERENCE:

http://www.iare.ac.in

PYTHON PROGRAMMING LABORATORY

<table>
<tr><td colspan="9">I Semester: Common from all branches</td></tr>
<tr><th>Course Code</th><th>Category</th><th colspan="3">Hours / Week</th><th>Credits</th><th colspan="3">Maximum Marks</th></tr>
<tr><td rowspan="2">ACSC02</td><td rowspan="2">Foundation</td><td>L</td><td>T</td><td>P</td><td>C</td><td>CIA</td><td>SEE</td><td>Total</td></tr>
<tr><td>0</td><td>0</td><td>3</td><td>1.5</td><td>30</td><td>70</td><td>100</td></tr>
<tr><td>Contact Classes: Nil</td><td>Tutorial Classes: Nil</td><td colspan="4">Practical Classes: 36</td><td colspan="3">Total Classes:36</td></tr>
<tr><td colspan="9">Prerequisite: There are no prerequisites to take this course.</td></tr>
</table>

I. COURSE OVERVIEW:

This course introduces students to writing computer programs. This course presents the principles of structured programming using the Python language, one of the most increasingly preferred languages for programming today. Because of its ease of use, it is ideal as a first programming language and runs on both the PC and Macintosh platforms. However, the knowledge gained in the course can be applied later to other languages such as C and Java. The course uses iPython Notebook to afford a more interactive experience. Topics include fundamentals of computer programming in Python, object-oriented programming and graphical user interfaces.

II. COURSE OBJECTIVES:

The students will try to learn:

VI. Acquire programming skills in core Python.
VII. Acquire Object-oriented programming skills in Python.
VIII. Develop the skill of designing graphical-user interfaces (GUI) in Python.
IX. Develop the ability to write database applications in Python.
X. Acquire Python programming skills to move into specific branches - Internet of Things (IoT), Data Science, Machine Learning (ML), Artificial Intelligence (AI) etc.

III. COURSE SYLLABUS:

Week – 1: OPERATORS

a. Read a list of numbers and write a program to check whether a particular element is present or not using membership operators.
b. Read your name and age and write a program to display the year in which you will turn 100 years old.
c. Read radius and height of a cone and write a program to find the volume of a cone.
d. Write a program to compute distance between two points taking input from the user (Hint: use Pythagorean theorem)

Week – 2: CONTROL STRUCTURES

a. Read your email id and write a program to display the no of vowels, consonants, digits and white spaces in it using if…elif…else statement.
b. Write a program to create and display a dictionary by storing the antonyms of words. Find the antonym of a particular word given by the user from the dictionary using while loop.
c. Write a Program to find the sum of a Series 1/1! + 2/2! + 3/3! + 4/4! +…….+ n/n!. (Input :n = 5, Output : 2.70833)
d. In number theory, an abundant number or excessive number is a number for which the sum of its proper divisors is greater than the number itself. Write a program to find out, if the given number is abundant. (Input: 12, Sum of divisors of 12 = 1 + 2 + 3 + 4 + 6 = 16, sum of divisors 16 > original number 12)

Week – 3: LIST

a. Read a list of numbers and print the numbers divisible by x but not by y (Assume x = 4 and y = 5).
b. Read a list of numbers and print the sum of odd integers and even integers from the list.(Ex: [23, 10, 15, 14, 63], odd numbers sum = 101, even numbers sum = 24)
c. Read a list of numbers and print numbers present in odd index position. (Ex: [10, 25, 30, 47, 56, 84, 96], The numbers in odd index position: 25 47 84).
d. Read a list of numbers and remove the duplicate numbers from it. (Ex: Enter a list with duplicate elements: 10 20 40 10 50 30 20 10 80, The unique list is: [10, 20, 30, 40, 50, 80])

Week – 4: TUPLE

a. Given a list of tuples. Write a program to find tuples which have all elements divisible by K from a list of tuples. test_list = [(6, 24, 12), (60, 12, 6), (12, 18, 21)], K = 6, Output : [(6, 24, 12), (60, 12, 6)]
b. Given a list of tuples. Write a program to filter all uppercase characters tuples from given list of tuples. (Input: test_list = [(“GFG”, “IS”, “BEST”), (“GFg”, “AVERAGE”), (“GfG”,), (“Gfg”, “CS”)], Output : [(‘GFG’, ‘IS’, ‘BEST’)]).
c. Given a tuple and a list as input, write a program to count the occurrences of all items of the list in the tuple. (Input : tuple = ('a', 'a', 'c', 'b', 'd'), list = ['a', 'b'], Output : 3)

Week – 5: SET

a. Write a program to generate and print a dictionary that contains a number (between 1 and n) in the form (x, x*x).
b. Write a program to perform union, intersection and difference using Set A and Set B.
c. Write a program to count number of vowels using sets in given string (Input : “Hello World”, Output: No. of vowels : 3)
d. Write a program to form concatenated string by taking uncommon characters from two strings using set concept (Input : S1 = "aacdb", S2 = "gafd", Output : "cbgf").

Week – 6: DICTIONARY

a. Write a program to do the following operations:
   i. Create a empty dictionary with dict() method
   ii. Add elements one at a time
   iii. Update existing key’s value
   iv. Access an element using a key and also get() method
   v. Deleting a key value using del() method
b. Write a program to create a dictionary and apply the following methods:
   i. pop() method
   ii. popitem() method
   iii. clear() method
c. Given a dictionary, write a program to find the sum of all items in the dictionary.
d. Write a program to merge two dictionaries using update() method.

Week – 7: STRINGS

a. Given a string, write a program to check if the string is symmetrical and palindrome or not. A string is said to be symmetrical if both the halves of the string are the same and a string is said to be a palindrome string if one half of the string is the reverse of the other half or if a string appears same when read forward or backward.
b. Write a program to read a string and count the number of vowel letters and print all letters except 'e' and 's'.
c. Write a program to read a line of text and remove the initial word from given text. (Hint: Use split() method, Input : India is my country. Output : is my country)
d. Write a program to read a string and count how many times each letter appears. (Histogram).

Week – 8: USER DEFINED FUNCTIONS

a. A generator is a function that produces a sequence of results instead of a single value. Write a generator function for Fibonacci numbers up to n.
b. Write a function merge_dict(dict1, dict2) to merge two Python dictionaries.
c. Write a fact() function to compute the factorial of a given positive number.
d. Given a list of n elements, write a linear_search() function to search a given element x in a list.

Week – 9: BUILT-IN FUNCTIONS

a. Write a program to demonstrate the working of built-in statistical functions mean(), mode(), median() by importing statistics library.
b. Write a program to demonstrate the working of built-in trignometric functions sin(), cos(), tan(), hypot(), degrees(), radians() by importing math module.
c. Write a program to demonstrate the working of built-in Logarithmic and Power functions exp(), log(), log2(), log10(), pow() by importing math module.

d. Write a program to demonstrate the working of built-in numeric functions ceil(), floor(), fabs(), factorial(), gcd() by importing math module.

Week – 10: CLASS AND OBJECTS

a. Write a program to create a BankAccount class. Your class should support the following methods for
   i) Deposit
   ii) Withdraw
   iii) GetBalanace
   iv) PinChange
b. Create a SavingsAccount class that behaves just like a BankAccount, but also has an interest rate and a method that increases the balance by the appropriate amount of interest (Hint:use Inheritance).
c. Write a program to create an employee class and store the employee name, id, age, and salary using the constructor. Display the employee details by invoking employee_info() method and also using dictionary (__dict__).
d. Access modifiers in Python are used to modify the default scope of variables. Write a program to demonstrate the 3 types of access modifiers: public, private and protected.

Week – 11: MISCELLANEOUS PROGRAMS

a. Write a program to find the maximum and minimum K elements in Tuple using slicing and sorted() method (Input: test_tup = (3, 7, 1, 18, 9), k = 2, Output: (3, 1, 9, 18))
b. Write a program to find the size of a tuple using getsizeof() method from sys module and built-in __sizeof__() method.
c. Write a program to check if a substring is present in a given string or not.
d. Write a program to find the length of a string using various methods:
   i. Using len() method
   ii. Using for loop and in operator
   iii. Using while loop and slicing

Week – 12: ADDITIONAL PROGRAMS - FILE HANDLING

1. Write a program to read a filename from the user, open the file (say firstFile.txt) and then perform the following operations:
   i. Count the sentences in the file.
   ii. Count the words in the file.
   iii. Count the characters in the file.
2. Create a new file (Hello.txt) and copy the text to other file called target.txt. The target.txt file should store only lower case alphabets and display the number of lines copied.
3. Write a Python program to store N student's records containing name, roll number and branch. Print the given branch student's details only.

IV. REFERENCE BOOKS:

1. Michael H Goldwasser, David Letscher, "Object Oriented Programming in Python", Prentice Hall, 1st Edition, 2007.
2. Yashavant Kanetkar, Aditya Kanetkar, "Let us Python", BPB publication, 1st Edition, 2019.
3. Ashok Kamthane, Amit Kamthane, "Programming and Problem Solving with Python", McGraw Hill Education (India) Private Limited, 2018.
4. Taneja Sheetal, Kumar Naveen, "Python Programming – A modular approach", Pearson, 2017.
5. R Nageswara Rao, "Core Python Programming", Dreamtech press, 2017 Edition.

V. WEB REFERENCES:

1. https://realpython.com/python3-object-oriented-programming/
2. https://python.swaroopch.com/oop.html
3. https://python-textbok.readthedocs.io/en/1.0/Object_Oriented_Programming.html
4. https://www.programiz.com/python-programming/
5. https://www.geeksforgeeks.org/python-programming-language/

CHEMISTRY

I Semester: CSE / CSE (AI&ML) / CSE (DS) / CSE (CS) / / CSIT / IT II Semester: AE / ME / CE / ECE / EEE								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
AHSC06	**Foundation**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		2	-	-	2	30	70	100
Contact Classes: 45	**Tutorial Classes: 0**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite: There are no prerequisites to take this course.								

I. COURSE OVERVIEW:

The concepts developed in this course involve elements and compounds and their applied industrial applications. It deals with topics such as batteries, corrosion and control of metallic materials, water and its treatment for different purposes, engineering materials such as plastics, elastomers and biodegradable polymers, their preparation, properties and applications, energy sources and environmental science. Sustainable chemistry that focuses on the design of the products and processes that minimize or eliminate the use and generation of hazardous substances is also included.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The concepts of electrochemical principles and causes of corrosion in the new developments and breakthroughs efficiently in engineering and technology.

II. The different parameters to remove causes of hardness of water and their reactions towards complexometric method.

III. The polymerization reactions with respect to mechanisms and its significance in industrial applications.

IV. The Significance of Green chemistry to reduce pollution in environment by using natural resources.

III. COURSE SYLLABUS

MODULE-I: ELECTROCHEMISTRY AND CORROSION (09)

Electro chemical cells: Electrode potential, standard electrode potential, Calomel electrode and Nernst equation; Electrochemical series and its applications; Numerical problems; Batteries: Primary (Dry cell) and secondary batteries (Lead-acid storage battery, Li-ion battery). Corrosion: Causes and effects of corrosion: Theories of chemical and electrochemical corrosion, mechanism of electrochemical corrosion; Corrosion control methods: Cathodic protection, sacrificial anode and impressed current Cathodic protection; Surface coatings: Metallic coatings- Methods of coating- Hot dipping- galvanization and tinning, electroplating.

MODULE –II: WATER TECHNOLOGY (09)

Introduction: Hardness of water, causes of hardness; types of hardness: temporary and permanent hardness, expression and units of hardness; estimation of hardness of water by complexometric method; potable water and its specifications, Steps involved in the treatment of water, disinfection of water by chlorination and ozonization; External treatment of water; Ion-exchange process; Desalination of water: Reverse osmosis, numerical problems.

MODULE-III: ENGINEERING MATERIALS (09)

Polymers-classification with examples, polymerization-addition, condensation and co-polymerization;
Plastics: Thermoplastics and thermosetting plastics; Compounding of plastics; Preparation, properties and applications of polyvinyl chloride, Teflon, Bakelite and Nylon-6, 6; Elastomers: Natural rubber, processing of natural rubber, vulcanization; Buna-s and Thiokol rubber; Biodegradable polymers.

Lubricants: characteristics of lubricants, mechanism of lubrication – thick film, thin film, extreme pressure lubrication, properties – flash and fire point, cloud and pour point, viscosity and oiliness of lubricants.

MODULE –IV: GREEN CHEMISTRY AND FUELS (09)

Introduction: Definition of green chemistry, methods of green synthesis: aqueous phase, microwave method, phase transfer catalyst and ultra sound method. Fuels: definition, classification of fuels ; Solid fuels: coal; analysis of coal: proximate and ultimate analysis; Liquid fuels: Petroleum and its refining; Gaseous fuels: Composition, characteristics and applications of LPG and CNG; Calorific value: Gross Calorific value(GCV) and Net Calorific value(NCV), numerical problems.

MODULE –V: NATURAL RESOURCES AND ENVIRONMENTAL POLLUTION (09)

Natural resources: Classification of resources, living and nonliving resources; Water resources: Use and over utilization of surface and ground water, floods and droughts, dams, benefits and problems; Land resources; Energy resources: renewable and non-renewable energy sources, use of alternate energy source. Environmental pollution: Causes, effects and control measures of air pollution, water pollution, soil pollution and noise pollution.

IV. TEXT BOOKS:

1. P. C. Jain and Monica Jain, "Engineering Chemistry", DhanpatRai Publishing Company, 16th Edition, 2017.
2. ShashiChawla, "Text Book of Engineering Chemistry" DhanatRai and Company, 2017.
3. Prashanthrath, B.Rama Devi, Ch.VenkataRamana Reddy, Subhendu Chakroborty, Cengage Learning Publishers, 1st Edition, 2018.

V. REFERENCE BOOKS:

1. Bharathi Kumari, "Engineering Chemistry", VGS Book Links, 10th Edition, 2018.
2. B. Siva Shankar, "Engineering Chemistry", Tata McGraw Hill Publishing Limited, 3rd Edition, 2015.
3. S. S. Dara, Mukkanti, "Text of Engineering Chemistry", S. Chand & Co, New Delhi, 12th Edition, 2006.

VI. WEB REFERENCES:

1. Engineering chemistry (NPTEL Web-book), by B.L.Tembe, Kamaluddin and M.S.Krishnan. http://www.cdeep.iitb.ac.in/webpage_data/nptel/Core%20Science/Engineering%20Chemistry%201/About-Faculty.html
2. Polymer Science (NPTEL Web-book), by Prof. Dibakar Dhara https://onlinecourses.nptel.ac.in/noc20_cy21/preview
3. Environmental Chemistry and Analysis(NPTEL Web-book), by Prof. M.S.Subramanian https://nptel.ac.in/courses/122/106/122106030/

MATHEMATICAL TRANSFORM TECHNIQUES

<table>
<tr><td colspan="10">II Semester: AE / ME / CE / ECE / EEE</td></tr>
<tr><th>Course Code</th><th>Category</th><th colspan="3">Hours / Week</th><th>Credits</th><th colspan="3">Maximum Marks</th></tr>
<tr><td rowspan="2">AHSC07</td><td rowspan="2">Foundation</td><td>L</td><td>T</td><td>P</td><td>C</td><td>CIA</td><td>SEE</td><td>Total</td></tr>
<tr><td>3</td><td>1</td><td>-</td><td>4</td><td>30</td><td>70</td><td>100</td></tr>
<tr><td>Contact Classes: 45</td><td>Tutorial Classes: 15</td><td colspan="4">Practical Classes: Nil</td><td colspan="3">Total Classes: 60</td></tr>
<tr><td colspan="9">Prerequisite: Basic principles of calculus</td></tr>
</table>

I. COURSE OVERVIEW:

This course focuses on transformations from theoretical based mathematical laws to its practical applications in the domain of various branches of engineering field. The course includes the transformations such as Laplace, Fourier, applications of scalar and vector field over surface, volume and multiple integrals. The course is designed to extract the mathematical developments, skills, from basic concepts to advance level of engineering problems to meet the technological challenges.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The transformation of ordinary differential equations in Laplace field and its applications.
II. The operation of the non-periodic functions by Fourier transforms.
III. The concepts of *multiple integration for finding* areas and volumes of physical quantities.
IV. The Integration of the several functions by transforming the co-ordinate system in scalar and vector fields.

III. COURSE SYLLABUS

MODULE-I: LAPLACE TRANSFORMS (09)

Definition of Laplace transform, linearity property, piecewise continuous function, existence of Laplace transform, function of exponential order, first and second shifting theorems, change of scale property, Laplace transforms of derivatives and integrals, multiplied by t, divided by t, Laplace transform of periodic functions.

Inverse Laplace transform: Definition of Inverse Laplace transform, linearity property, first and second shifting theorems, change of scale property, multiplied by s, divided by s; Convolution theorem and applications to ordinary differential equations.

MODULE –II: FOURIER TRANSFORMS (09)

Fourier integral theorem, Fourier sine and cosine integrals; Fourier transforms; Fourier sine and cosine transform, properties, inverse transforms, finite Fourier transforms. .

MODULE –III: MULTIPLE INTEGRALS (09)

Double Integrals: Evaluation of double integrals in Cartesian coordinates and Polar coordinates; Change of order of integration; Area as a double integral; Transformation of coordinate system.

Triple Integrals: Evaluation of triple integrals in Cartesian coordinates; volume of a region using triple integration.

MODULE –IV: VECTOR DIFFERENTIAL CALCULUS (09)

Scalar and vector point functions; Definitions of Gradient, divergent and curl with examples; Solenoidal and irrotational vector point functions; Scalar potential function. Line integral, surface integral and volume integral, Green‘s theorem in a plane, Stoke‘s theorem and Gauss divergence theorem without proofs.

MODULE –V: PARTIAL DIFFERENTIAL EQUATIONS (09)

Formation of partial differential equations by elimination of arbitrary constants and arbitrary functions, solutions of first order linear equations; Charpit’s method;

IV. TEXT BOOKS:

1. B.S. Grewal, “Higher Engineering Mathematics”, Khanna Publishers, 36th Edition, 2010.
2. N.P. Bali and Manish Goyal, “A text book of Engineering Mathematics”, Laxmi Publications, Reprint, 2008.
3. Ramana B.V., “Higher Engineering Mathematics”, Tata McGraw Hill New Delhi, 11th Reprint, 2010

V. REFERENCE BOOKS:

1. Erwin Kreyszig, “Advanced Engineering Mathematics”, John Wiley & Sons, 9thEdition, 2006.
2. Veerarajan T., “Engineering Mathematics for first year”, Tata McGraw-Hill, New Delhi, 2008.
3. D. Poole, “Linear Algebra: A Modern Introduction”, Brooks/Cole, 2ndEdition, 2005.
4. Dr. M Anita, “Engineering Mathematics-I”, Everest Publishing House, Pune, First Edition, 2016.

VI. WEB REFERENCES:

1. http://www.efunda.com/math/math_home/math.cfm
2. http://www.ocw.mit.edu/resources/#Mathematics
3. http://www.sosmath.com
4. http://www.mathworld.wolfram.com.

VII. E-TEXT BOOKS:

1. http://www.e-booksdirectory.com/details.php?ebook=10166
2. http://www.e-booksdirectory.com/details.php?ebook=7400re

ELECTRICAL CIRCUITS

II Semester: EEE / ECE								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
AEEC02	**Foundation**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		3	-	-	3	30	70	100
Contact Classes: 45	**Tutorial Classes: Nil**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisites: **There are no prerequisites to take this course.**								

I. COURSE OVERVIEW:

The course introduces the basic concepts of circuit analysis which is the foundation for all subjects of the electrical and electronics engineering. It includes the basic fundamental laws of electricity and magnetism with an emphasis on resistors, inductors and capacitors (RLC) circuits applied to alternating current (AC) or direct current (DC) of electrical networks. Further This course provides network theorems with different excitations, two-port network and network topology to solve for real-time applications.

II. COURSE OBJECTIVES:

The students will try to learn

I. The network reduction techniques such as source transformation, mesh analysis, nodal analysis and network theorems to solve different networks.
II. The basic concept of AC circuits for optimization of household and industrial circuitry.
III. The various configurations of electromagnetic induction used in magnetic circuits helps in the winding of electrical machines.
IV. The characteristics of two-port networks and network topologies suitable in power system.

III. COURSE OBJECTIVES:

MODULE-I: INTRODUCTION TO ELECTRICAL CIRCUITS (09)

Circuit concept: Basic definitions, Ohm's law at constant temperature, classifications of elements, independent and dependent sources, voltage and current relationships for passive elements,

Single phase AC circuits: Representation of alternating quantities, properties of different periodic wave forms, phase and phase difference, concept of impedance and admittance, power in AC circuits.

MODULE-II: ANALYSIS OF ELECTRICAL CIRCUITS (09)

Circuit analysis: Source transformation, Kirchhoff's laws, total resistance, inductance and capacitance of circuits, Star - delta transformation technique, mesh analysis and nodal analysis, inspection method, super mesh, super node analysis.

MODULE-III: NETWORK THEOREMS (DC AND AC) (10)

Network Theorems: Tellegen's, superposition, reciprocity, Thevenin's, Norton's, maximum power transfer, Milliman's and compensation theorems for DC excitations, numerical problems.

Network Theorems: Tellegen's, superposition, reciprocity, Thevenin's, Norton's, maximum power transfer, Milliman's and compensation theorems for AC excitations, numerical problems.

MODULE-IV: MAGNETIC CIRCUITS (09)

Magnetic circuits: Faraday's laws of electromagnetic induction, concept of self and mutual inductance, dot convention, coefficient of coupling, composite magnetic circuit, analysis of series and parallel magnetic circuits.

MODULE-V: **TWO PORT NETWORK AND GRAPH THEORY (08)**

Two Port Network: Two port parameters, interrelations, Two port Interconnections.

Network topology: Definitions, incidence matrix, basic tie set and basic cut set matrices for planar networks, duality and dual networks.

IV. TEXT BOOKS:

1. A Sudhakar, Shyammohan S Palli, "Circuits and Networks", Tata McGraw-Hill, 4th Edition, 2010.
2. M E Van Valkenberg, "Network Analysis", PHI, 3rd Edition, 2014.

V. REFERENCE BOOKS:

1. John Bird, "Electrical Circuit Theory and Technology", Newnes, 2nd Edition, 2003.
2. C L Wadhwa, "Electrical Circuit Analysis including Passive Network Synthesis", New Age International, 2nd Edition, 2009.
3. David A Bell, "Electric circuits", Oxford University Press, 7th Edition, 2009.
4. E Hughes, "Electrical and Electronics Technology", Pearson Education, 2010.
5. A Chakrabarthy, "Electric Circuits", Dhanipat Rai & Sons, 6th Edition, 2010.
6. V D Toro, "Electrical Engineering Fundamentals", Prentice Hall India, 1989.

VI. WEB REFERENCES:

1. https://www.igniteengineers.com
2. https://www.ocw.nthu.edu.tw
3. https://www.uotechnology.edu.iq
4. https://www.iare.ac.in

VII. E-TEXT BOOKS:

1. https://www.bookboon.com/en/concepts-in-electric-circuits-ebook
2. https://www.www.jntubook.com
3. https://www.allaboutcircuits.com
4. https://www.archive.org

PROGRAMMING FOR PROBLEM SOLVING USING C

<table>
<tr><td colspan="10">II Semester: CSE / CSE (AI & ML) / CSE (DS) / CSE (CS) / CSIT / IT / ECE / EEE</td></tr>
<tr><th>Course Code</th><th>Category</th><th colspan="3">Hours / Week</th><th>Credits</th><th colspan="3">Maximum Marks</th></tr>
<tr><td rowspan="2">ACSC04</td><td rowspan="2">Foundation</td><td>L</td><td>T</td><td>P</td><td>C</td><td>CIA</td><td>SEE</td><td>Total</td></tr>
<tr><td>3</td><td>-</td><td>-</td><td>3</td><td>30</td><td>70</td><td>100</td></tr>
<tr><td>Contact Classes: 45</td><td>Tutorial Classes: Nil</td><td colspan="4">Practical Classes: Nil</td><td colspan="3">Total Classes: 45</td></tr>
<tr><td colspan="10">Prerequisite: There are no prerequisites to take this course.</td></tr>
</table>

I. COURSE OVERVIEW:

The main emphasis of the course will be on problem solving aspects in through C programming. The students will understand programming language, programming, concepts of loops, reading a set of data, step wise refinements, functions, control structures, arrays, dynamic memory allocations, enumerated data types, structures, unions, and file handling. This course provides adequate knowledge to solve problems in their respective domains.

II. COURSE OBJECTIVES:

The students will try to learn:

I. Problem-solving through programming
II. Programming language, programming, reading a set of Data, stepwise refinement, concepts of Loops, Functions, Control structure, Arrays, Structure, Pointer and File concept.
III. To build efficient programs in 'C' language essential for future programming and software engineering courses.

III. COURSE SYLLABUS:

MODULE-I: INTRODUCTION (10)

Introduction to components of a computer: Memory, processor, I/O Devices, storage, operating system; Concept of assembler, compiler, interpreter, loader and linker.

Idea of Algorithms: Algorithms, Flowcharts, Pseudo code with examples, From algorithm to Programs and Source Code

Introduction to C Programming Language: History of C, Basic structure of a C program, Process of compiling and running a C program; C Tokens: Keywords, Identifiers, Constants, Strings, Special symbols, Variables, Data types; Operators, Precedence of Operators, Expression evaluation, Formatted Input/Output functions, Type Conversion and type casting.

MODULE-II: CONTROL STRUCTURES (08)

Decision Making Statements: Simple if, if-else, else if ladder, Nested if, switch case statement;
Loop control statements: for, while and do while loops, nested loops;
Unconditional Control Structures: break, continue and goto statements.

MODULE-III: ARRAYS AND FUNCTIONS (10)

Arrays: Introduction, Single dimensional array and multi-dimensional array: declaration, initialization, accessing elements of an array; Operations on arrays: traversal, reverse, insertion, deletion, merge, search; **Strings:** Arrays of characters, Reading and writing strings, String handling functions, Operations on strings; array of strings.

Functions: Concept of user defined functions, Function declaration, return statement, Function prototype, Types of functions, Inter function communication, Function calls, Parameter passing mechanisms; Recursion; Passing arrays to functions, passing strings to functions; Storage classes.

MODULE-IV: POINTER AND STRUCTURES (10)

Pointers: Basics of pointers, Pointer arithmetic, pointer to pointers, array of pointers, Generic pointers, Null pointers, Pointers as functions arguments, Functions returning pointers; Dynamic memory allocation.
Structures: Structure definition, initialization, structure members, nested structures, arrays of structures, structures and functions, structures and pointers, self-referential structures; Unions: Union definition, initialization, accessing union members; bit fields, typedef, enumerations, Preprocessor directives.

MODULE-V: FILE HANDLING AND APPLICATIONS IN C (07)

File Handling: Concept of a file, text files and binary files, streams, standard I/O, formatted I/O, file I/O operations, error handling, Line I/O, miscellaneous functions; Applications in C.

IV.TEXT BOOKS:

1. Byron Gottfried, "Programming with C", Schaum's Outlines Series, McGraw Hill Education, 3rd Edition, 2017.
2. Reema Thareja, "Programming in C", Oxford university press, 2nd Edition, 2016.

V. REFERENCE BOOKS:

I. W. Kernighan Brian, Dennis M. Ritchie, "The C Programming Language", PHI Learning, 2nd Edition, 1988.
II. Yashavant Kanetkar, "Exploring C", BPB Publishers, 2nd Edition, 2003.
III. Schildt Herbert, "C: The Complete Reference", Tata McGraw Hill Education, 4th Edition, 2014.
IV. R. S. Bichkar, "Programming with C", Universities Press, 2nd Edition, 2012.
V. Dey Pradeep, Manas Ghosh, "Computer Fundamentals and Programming in C", Oxford University Press, 2nd Edition, 2006.
VI. Stephen G. Kochan, "Programming in C", Addison-Wesley Professional, 4th Edition, 2014.

VI.WEB REFERENCES:

1. https://www.calvin.edu/~pribeiro/courses/engr315/EMFT_Book.pdf
2. https://www.web.mit.edu/viz/EM/visualizations/coursenotes/modules/guide02.pdf
3. https://www.nptel.ac.in/courses/108106073/
4. https://www.iare.ac.in

VII.E-TEXT BOOKS:

1. http://www.freebookcentre.net/Language/Free-C-Programming-Books-Download.htm
2. http://www.imada.sdu.dk/~svalle/courses/dm14-2005/mirror/c/
3. http://www.enggnotebook.weebly.com/uploads/2/2/7/1/22718186/ge6151-notes.pdf

EXPERIENTIAL ENGINEERING EDUCATION (ExEEd) - ACADEMIC SUCCESS

I Semester: CSE / CSE (AI&ML) / CSE (DS) / CSE(CS) / IT / CSIT

II Semester: AE / ME / CE / ECE / EEE

Course Code	Category	Hours / Week			Credits	Maximum Marks		
ACSC06	Foundation	L	T	P	C	CIA	SEE	Total
		2	-	-	1	30	70	100
Contact Classes: Nil	Tutorial Classes: Nil	Practical Classes: 36				Total Classes: 36		

Prerequisite: There are no prerequisites to take this course

I.COURSE OVERVIEW:

The course aims to provide students with an understating of the different learning –coding platforms, role of the entrepreneur, innovation and technology in customer centric engineering.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The different ways in engaging continuous learning process through the interaction with peers in related topics.
II. The skills and potential opportunities using well know frameworks and analytical tools.
III. The attitudes, values, characteristics, behavior and processes with processing an entrepreneurial mindset.

III. COURSE OBJECTIVES:

WEEK – I
Introduction to ExEED - Dr. Ch. Srinivasulu

WEEK – II:
Skill Development - Dr. G Ramu

WEEK – III:
Skill Development - Dr. G Ramu

WEEK – IV:
Open Source platforms for Learning , Practice and Excel in their field - Dr. M MadhuBala

WEEK – V:
Opportunities and challenges - Respective Department HOD's

WEEK – VI:
Skill Development - Dr. G Ramu

WEEK – VII:
Skill Development - Dr. G Ramu

WEEK –VIII:
Entrepreneurial Mindset - Dr. J Sirisha Devi

WEEK – IX:
Entrepreneurial Mindset - Dr. J Sirisha Devi

WEEK – X:
Innovation Culture - Dr. M Pala Prasad Reddy

WEEK – XI:
Support & Funding from various organizations - Dr. M Pala Prasad Reddy

WEEK – XII:
Rapid Prototyping - Prof. V V S H Prasad

WEEK – XIII:
Intellectual Property Rights - Mr. K Aditya Nag

WEEK – XIV:
Story Telling by Students - Dr. Ch. Srinivasulu

ELECTRICAL CIRCUITS LABORATORY

II Semester: ECE / EEE								
Course Code	Category	Hours / Week			Credits	Maximum Marks		
AEEC03	Foundation	L	T	P	C	CIA	SEE	Total
		-	-	3	1.5	30	70	100
Contact Classes: Nil	Tutorial Classes: Nil	Practical Classes: 36				Total Classes: 36		
Prerequisites: There are no prerequisites to take this course.								

I. COURSE OVERVIEW:

Electrical circuits laboratory examines the basic laws, network reduction techniques, network theorems, characteristics of AC and two port network, design of transformer, measurement of electrical parameters. It includes the basic concepts of MATLAB.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The basic law, network reduction techniques and network theorems for circuit analysis.
II. The characteristics of AC and two port networks.
III. The measurement of electrical quantities using various electrical devices.
IV. To built the prototype of transformer and study its properties.

III. COURSE SYLLABUS:

Expt. 1: VERIFICATION OF OHM'S LAW AND KIRCHOFF LAWS
Draw the V-I characteristics of resistor element, examine voltage and current division in an electrical circuit using hardware and digital simulation.

Expt. 2: MESH ANALYSIS
Determination of mesh currents in complex electrical circuit using hardware and digital simulation.

Expt. 3: NODAL ANALYSIS
Determination of nodal voltages in complex electrical circuit using hardware and digital simulation.

Expt. 4: CHARECTERISTICS OF PERIODIC WAVEFORMS
Calculate Instantaneous, Peak, Peak to peak, Average and RMS values of periodic wave form using hardware and digital simulation.

Expt. 5: DETERMINATION OF CIRCUIT IMPEDANCE
Find the impedance of series RL, RC and RLC circuits using hardware and digital simulation.

Expt. 6: THEVENIN'S THEOREM
Determine load or unknown current using Thevenin's equivalent circuit using hardware and digital simulation.

Expt. 7: NORTON'S THEOREM
Determine load or unknown current using Norton's equivalent circuit using hardware and digital simulation.

Expt. 8: SUPERPOSITION THEOREM
Verify of superposition theorem using hardware and digital simulation.

Expt. 9: RECIPROCITY THEOREM
Verify of reciprocity theorem using hardware and digital simulation.

Expt. 10: SERIES AND PARALLEL RESONANCE
Verification of series and parallel resonance using hardware and digital simulation.
Expt. 11: MEASUREMENT OF POWER CONSUMED BY A FLUORESCENT LAMP

Examine the power consumed by Fluorescent lamp using electrical devices using hardware and digital simulation.

Expt. 12: **DESIGN OF CHOKE AND SMALL TRANSFORMER**
Measure resistance and inductance of coil and construct the winding of transformer using winding machine using hardware and digital simulation.

Expt. 13: Z AND Y PARAMETERS
Determine the open circuit and short circuit parameters for two port network using hardware and digital simulation.

Expt. 14: **H AND ABCD PARAMETERS**
Determine the hybrid and transmission line parameters for two port network using hardware and digital simulation.

IV. REFERENCE BOOKS:

1. A Chakrabarti, “Circuit Theory”, Dhanpat Rai Publications, 6th Edition, 2006.
2. William Hayt, Jack E Kemmerly S.M. Durbin, “Engineering Circuit Analysis”, Tata McGraw Hill, 7th Edition, 2010.
3. K S Suresh Kumar, “Electric Circuit Analysis”, Pearson Education, 1st Edition, 2013.

V. WEB REFERENCES:

1. https://www.ee.iitkgp.ac.in
2. https://www.citchennai.edu.in
3. https://www.iare.ac.in

PROGRAMMING FOR PROBLEM SOLVING USING C LABORATORY

II Semester: CSE / CSE (AI & ML) / CSE (DS) / CSE (CS) / CSIT / IT / ECE /EEE

Course Code	Category	Hours / Week			Credits	Maximum Marks		
		L	T	P	C	CIA	SEE	Total
ACSC05	Foundation	0	0	3	1.5	30	70	100
Contact Classes: Nil	Tutorial Classes: Nil	Practical Classes: 36				Total Classes:36		

Prerequisite: There are no prerequisites to take this course.

I. COURSE OVERVIEW

This course introduces students to writing computer programs. This course presents the principles of structured programming using the Python language, one of the most increasingly preferred languages for programming today. Because of its ease of use, it is ideal as a first programming language and runs on both the PC and Macintosh platforms. However, the knowledge gained in the course can be applied later to other languages such as C and Java. The course uses iPython Notebook to afford a more interactive experience.

II. COURSE OBJECTIVES:

The students will try to learn:

I. Acquire logical thinking and identify efficient ways of solving problems using C programming language.
II. Develop programs by using decision making, branching and looping constructs.
III. Implement real time applications using the concept of array, pointers, functions and structures.

III. COURSE SYLLABUS:

Week – 1: OPERATORS AND EVALUATION OF EXPRESSIONS

e. Design and develop a flowchart and algorithm to read a number and implement using a C program to check whether the given number is even or odd using ternary operator.
f. Design and develop a flowchart and algorithm to read two integers and implement using a C program to perform the addition of two numbers without using + operator.
g. Develop a C program to evaluate the following arithmetic expressions by reading appropriate input from the standard input device. Understand the priority of operators while evaluating expressions.
   i. 6*2/(2+1 * 2/3 +6) +8 * (8/4)
   ii. 17 – 8 / 4 * 2 + 3 - ++2
   iii. ! (x > 10) && (y = = 2)
h. Develop a C program to display the size of various built-in data types in C language.

Week – 2: CONTROL STRUCTURES

a. Design and develop a flowchart and algorithm to read a year as an input and find whether it is leap year or not. Implement a C program for the same and execute for all possible inputs with appropriate messages. Also consider end of the centuries.
b. Design and develop a flowchart and algorithm to find the square root of a given number N. Implement a C program for the same and execute for all possible inputs with appropriate messages. (Note: Don't use library function sqrt(n), Hint: Use Newton-Raphson method to find the square root).
c. Design and develop a flowchart and algorithm to generate a Fibonacci sequence up to a given number N. A Fibonacci sequence is defined as follows: The first and second terms in the sequence are 0 and 1. Subsequent terms are found by adding the preceding two terms in the sequence. Implement a C program for the developed flowchart/algorithm and execute the same to generate the first N terms of the sequence.
d. Design and develop a flowchart and algorithm that takes three coefficients (a, b, and c) of a Quadratic equation ($ax^2+bx+c=0$) as input and compute all possible roots. Implement a C program for the developed flowchart/algorithm and execute the same to output the possible roots for a given set of coefficients with appropriate messages.

Week – 3: CONTROL STRUCTURES

a. Design and develop an algorithm to find the reverse of an integer number N and check whether it is PALINDROME or NOT. Implement a C program for the developed algorithm that takes an integer number as input and output the reverse of the same with suitable messages. Ex: N: 2020, Reverse: 0202, Not a Palindrome.

b. Draw the flowchart and write C Program to compute sin(x) using Taylor series approximation given by

$$\sin(x) = x - (x^3/3!) + (x^5/5!) - (x^7/7!) + \ldots\ldots$$

Compare the result with the built- in Library function and print both the results with appropriate messages.

c. Design and develop an algorithm and flowchart to read a three digit number and check whether the given number is Armstrong number or not. Write a C program to implement the same and also display the Armstrong numbers between the ranges 1 to 1000.

d. Design and develop an algorithm for evaluating the polynomial $f(x) = a_4x^4 + a_3x^3 + a_2x^2 + a_1x^1 + a_0$, for a given value of x and its coefficients using Horner's method. Implement a C program for the same and execute the program for different sets of values of coefficients and x.

Week – 4: ARRAYS

e. Develop, implement and execute a C program to read a list of integers and store it in a single dimensional array. Write a C program to print the second largest integer in a list of integers.

f. Develop, implement and execute a C program to read a list of integers and store it in a single dimensional array. Write a C program to count and display positive, negative, odd and even numbers in an array.

g. Develop, implement and execute a C program to read a list of integers and store it in a single dimensional array. Write a C program to find the frequency of a particular number in a list of integers.

h. Develop, implement and execute a C program that reads two matrices A (m x n) and B (p x q) and Compute the product A and B. Read matrix A and matrix B in row major order respectively. Print both the input matrices and resultant matrix with suitable headings and output should be in matrix format only. Program must check the compatibility of orders of the matrices for multiplication. Report appropriate message in case of incompatibility.

Week – 5: STRINGS

a. Develop a user-defined function **STRCOPY (str1, str2)** to simulate the built-in library function **strcpy (str1, str2)** that copies a string str2 to another string str1. Write a C program that invokes this function to perform string copying. Also perform the same operation using built-in function.

b. Develop a user-defined function **STRCONCT (str1, str2)** to simulate the built-in library function **strcat (str1, str2)** that takes two arguments str1 and str2, concatenates str2 and str1 and stores the result in str1. Write a C program that invokes this function to perform string concatenation. Also perform the same operation using built-in function.

c. Develop a C program that returns a pointer to the first occurrence of the string in a given string using built-in library function **strstr()**. Example: **strstr()** function is used to locate first occurrence of the string "test" in the string "This is a test string for testing". Pointer is returned at first occurrence of the string "test".

d. Develop a C program using the library function **strcmp (str1, str2)** that compares the string pointed to by str1 to the string pointed to by str2 and returns an integer. Display appropriate messages based on the return values of this function as follows −

if return value < 0 then it indicates str1 is less than str2.
if return value > 0 then it indicates str2 is less than str1.
if return value = 0 then it indicates str1 is equal to str2.

Week – 6: FUNCTIONS

a. Design and develop a recursive and non-recursive function **FACT(num)** to find the factorial of a number, n!, defined by FACT(n) = 1, if n = 0. Otherwise FACT(n) = n * FACT(n-1). Using this function, write a C program to compute the binomial coefficient. Tabulate the results for different values of n and r with suitable messages

b. Design and develop a recursive function **GCD (num1, num2**) that accepts two integer arguments. Write a C program that invokes this function to find the greatest common divisor of two given integers.

c. Design and develop a recursive function **FIBO (num)** that accepts an integer argument. Write a C program that invokes this function to generate the Fibonacci sequence up to num.

d. Design and develop a C function **ISPRIME** (num) that accepts an integer argument and returns 1 if the argument is prime, a 0 otherwise. Write a C program that invokes this function to generate prime numbers between the given ranges.

e. Design and develop a function **REVERSE (str**) that accepts a string arguments. Write a C program that invokes this function to find the reverse of a given string.

Week – 7: POINTERS

a. Develop a C program using pointers to compute the sum, mean and standard deviation of all elements stored in an array of n real numbers.
b. Develop a C program to read a list of integers and store it in an array. Then read the array elements using a pointer and print the value along with the memory addresses.
c. Design and develop non-recursive functions **input_matrix(matrix, rows, col**s) and **print_matrix(matrix, rows, cols)** that stores integers into a two-dimensional array and displays the integers in matrix form. Write a C program to input and print elements of a two dimensional array using pointers and functions.
d. Develop a C program to a store a list of integers in a single dimensional array using dynamic memory allocation (limit will be at run time) using malloc() function. Write a C program to read the elements and print the sum of all elements along with the entered elements. Also use free() function to release the memory.

Week – 8: **STRUCTURES AND UNIONS**

e. Write a C program that uses functions to perform the following operations:
   i. Reading a complex number
   ii. Writing a complex number
   iii. Addition and subtraction of two complex numbers
   Note: represent complex number using a structure.
f. Write a C program to compute the monthly pay of 100 employees using each employee_s name, basic pay. The DA is computed as 52% of the basic pay. Gross-salary (basic pay + DA). Print the employees name and gross salary.
g. Create a Book structure containing book_id, title, author name and price. Write a C program to pass a structure as a function argument and print the book details.
h. Create a union containing 6 strings: name, home_address, hostel_address, city, state and zip. Write a C program to display your present address.

Week – 9: ADDITIONAL PROGRAMS

a. Write a C program to read in two numbers, x and n, and then compute the sum of this geometric progression: $1+x+x^2+x^3+\ldots\ldots\ldots+x^n$. For example: if n is 3 and x is 5, then the program computes 1+5+25+125. Print x, n, the sum. Perform error checking. For example, the formula does not make sense for negative exponents – if n is less than 0. Have your program print an error message if n<0, then go back and read in the next pair of numbers of without computing the sum. Are any values of x also illegal? If so, test for them too.
b. Develop a C program to find the 2's complement of a given binary number. 2's complement is obtained by scanning it from right to left and complementing all the bits after the first appearance of a 1. Thus 2's complement of 11100 is 00100. Write a C program to find the 2's complement of a binary number.
c. Develop a C program to convert a Roman numeral to its decimal equivalent. E.g. check for the inputs - Roman number IX is equivalent to 9 and Roman number XI is equivalent to 11.

Week – 10: PREPROCESSOR DIRECTIVES

a. Define a macro with one parameter to compute the volume of a sphere. Write a C program using this macro to compute the volume for spheres of radius 5, 10 and 15meters.
b. Define a macro that receives an array and the number of elements in the array as arguments. Write a C program for using this macro to print the elements of the array.
c. Write symbolic constants for the binary arithmetic operators +, -, *, and /. Write a C program to illustrate the use of these symbolic constants.

Week – 11: FILES

a. Create an employee file **employee.txt** and write 5 records having employee name, designation, salary, branch and city. Develop a C program to display the contents of **employee.txt** file.
b. Create a **studentolddata.txt** file containing student name, roll no, branch, section, address. Develop a C program to copy the contents of **studentolddata.txt** file to another file **studentnewdata.txt**.
c. Develop a C program to create a text file **info.txt** to store the information given below. Implement using a C program to count the number of words and characters in the file **info.txt**.

Test Data:
Input the file name to be opened : info.txt
Expected Output:
The content of the file info.txt are :
Welcome to IARE
Welcome to Computer Programming

The number of words in the file info.txt are : 7
The number of characters in the file info.txt are : 46

d. Given two university information files "**studentname.txt**" and "**roll_number.txt**" that contains students Name and Roll numbers respectively. Write a C program to create a new file called "**output.txt**" and copy the content of files "**studentname.txt**" and "**roll_number.txt**" into output file. Display the contents of output file "**output.txt**" on to the screen.

studname.txt	**roll_number.txt**
Asha	20951A1201
Bharath	20951A0502
Uma	20951A0456
Shilpa	20951A0305

Week – 12: **COMMAND LINE ARGUMENTS**

a. Develop a C program to read a set of arguments and display all arguments given through command line.
b. Develop a C program to read a file at command line argument and display the contents of the file.
c. Develop a C program to read N integers and find the sum of N integer numbers using command line arguments.
d. Develop a C program to read three integers and find the largest integer among three using command line argument.

IV. REFERENCE BOOKS:

1. Yashavant Kanetkar, "Let Us C", BPB Publications, New Delhi, 13th Edition, 2012.
2. Oualline Steve, "Practical C Programming", O'Reilly Media, 3rd Edition, 1997.
3. King KN, "C Programming: A Modern Approach", Atlantic Publishers, 2nd Edition, 2015.
4. Kochan Stephen G, "Programming in C: A Complete Introduction to the C Programming Language", Sam's Publishers, 3rd Edition, 2004.
5. Linden Peter V, "Expert C Programming: Deep C Secrets", Pearson India, 1st Edition, 1994.

V. WEB REFERENCES:

1. http://www.sanfoundry.com/c-programming-examples
2. http://www.geeksforgeeks.org/c
3. http://www.cprogramming.com/tutorial/c
4. http://www.cs.princeton.edu

ENGINEERING WORKSHOP PRACTICE

I Semester: CSE / CSE (AI&ML) / CSE (DS) / CSE (CS) / CSIT / IT **II Semester:** ECE / EEE								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
AMEC04	**Foundation**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		-	-	2	1	30	70	100
Contact Classes: Nil	**Tutorial Classes: Nil**	**Practical Classes: 28**				**Total Classes: 28**		
Prerequisite: **There are no prerequisites to take this course.**								

I. COURSE OVERVIEW:

The course is intended to provide the basic concepts about Engineering tools for cutting and measuring used in a workshop. The students will be benefited from hands on training process as well as knowledge to carry out a particular process for making a product. This course provides wider perspective of manufacturing, processes to learn and introduces major trades as well as digital manufacturing facilities.

II.COURSE OBJECTIVES:

The students will try to learn:

I. The application of jigs and fixtures, measuring, marking and cutting tools in various types of manufacturing processes.
II. The preparation of different joints in carpentry and fitting and also familiarizes wood working machinery.
III. The concepts of forming processes by forging, black-smithy and tin-smithy with an application extracts of Engineering Drawing.
IV. The standard electrical wiring practices for domestic and industrial appliances.
V. The current advancements in developing the prototype models through digital manufacturing facilities.

III. COURSE SYLLABUS:

Week-1: CARPENTRY-I
Batch I: Preparation of Tenon joint as per given dimensions.
Batch II: Preparation of Mortise joint as per given taperangle.

Week -2: CARPENTRY-II
Batch I: Preparation of dove tail joint as per given taper angle.
Batch II: Preparation of lap joint as per given dimensions.

Week-3: FITTING - I
Batch I: Make a straight fit for given dimensions.
Batch II: Make a square fit for given dimensions.

Week-4: FITTING - II
Batch I : Make a V fit for given dimensions
Batch II: Make a semicircular fit for given dimensions.

Week-5: BLACKSMITHY- I
Batch I: Prepare S-bend for given MS rod using open hearth furnace.
Batch II: Prepare J-bend for given MS rod using open hearth furnace.

Week-6: BLACKSMITHY- II
Batch I: Prepare Fan hook for given dimensions.
Batch II: Prepare Round to Square for given dimensions

Week-7: MOULD PREPARATION
Batch I: Prepare a wheel flange mould using a given wooden pattern.
Batch II: Prepare a bearing housing using an aluminum pattern.

Week-8: MOULD PREPARATION
Batch I: Prepare a bearing housing using an aluminum pattern.
Batch II: Prepare a wheel flange mould using a given wooden pattern.

Week-9: TINSMITHY- I
Batch I: Prepare the development of a surface and make a rectangular tray for given dimensions.
Batch II: Prepare the development of a surface and make a round tin for given dimensions.

Week-10: TINSMITHY- II
Batch I: Prepare the development of a surface and make a Square Tin, for given dimensions.
Batch II: Prepare the development of a surface and make a Conical Funnel for given dimensions.

Week-11: ELECTRICAL WIRING-I
Batch I: Make an electrical connection of two bulbs connected in series.
Batch II:Make an electrical connection of two bulbs connected in parallel

Week-12: ELECTRICAL WIRING-II
Batch I: Make an electrical connection of one bulb controlled by two switches connected.
Batch II: Make an electrical connection of tube light.

IV. REFERENCE BOOKS:

1. Hajra Choudhury S.K., Hajra Choudhury A.K. and Nirjhar Roy S.K., “Elements of Workshop Technology”, Media promoters and publishers private limited, Mumbai, Vol. I 2008 and Vol. II 2010.
2. Kalpakjian S, Steven S. Schmid, “Manufacturing Engineering and Technology”, Pearson Education India Edition, 4th Edition, 2002.
3. Gowri P. Hariharan, A. Suresh Babu, “Manufacturing Technology – I”, Pearson Education, 2008.
4. Roy A. Lindberg, “Processes and Materials of Manufacture”, Prentice Hall India, 4th Edition, 1998.
5. Rao P.N., “Manufacturing Technology”, Vol. I and Vol. II, Tata McGraw-Hill House, 2017.

V. WEB REFERENCES:

http://www.iare.ac.in

NETWORK ANALYSIS

III Semester: EEE								
Course Code	Category	Hours / Week			Credits	Maximum Marks		
AEEC05	Core	L	T	P	C	CIA	SEE	Total
		3	0	0	3	30	70	100
Contact Classes: 45	Tutorial Classes: Nil	Practical Classes: Nil				Total Classes: 45		
Prerequisite: Electrical Circuits								

I. COURSE OVERVIEW:

This course introduces the basic concepts of net work theory which is the foundation for all subjects of the electrical engineering discipline. The emphasis of this course is laid on the basic analysis of circuits which includes three phase circuits, transient analysis of DC and AC circuits, network functions, and two port net work parameters, Fourier analysis of AC circuits, design and analysis of filters.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The three phase systems for star and delta connected systems and perform three phase power calculations for balanced and unbalanced loads
II. Present the necessary mathematical background for the transient analysis of DC & AC circuits and study the transients using differential equation and Laplace transform approach for series and parallel circuits.
III. The concept of locus diagram for series and parallel circuits and discuss network functions and the stability criteria
IV. The steady state response of complex electrical circuits with Ac supply and application of concept of electrical resonance
V. Classify and design different types of filters and study their characteristics.

III. COURSE SYLLABUS:

MODULE-I: ANALYSIS OF AC CIRCUITS (10)

Steady state analysis: Steady state analysis of RL, RC and RLC circuits (in series, parallel and series parallel combinations) with sinusoidal excitation; Resonance: Series and parallel resonance, concept of band width and Q factor.

MODULE-II: SOLUTION OF FIRST AND SECOND ORDER NETWORKS (08)

Transient response: Initial conditions, transient response of RL, RC and RLC series and parallel circuits with DC and AC excitations, differential equation and Laplace transform approach.

MODULE-III: LOCUS DIAGRAMS AND NETWORKS FUNCTIONS (10)

Locus diagrams: Locus diagrams of RL, RC, RLC circuits.

Network Functions: The concept of complex frequency, physical interpretation, transform impedance, series and parallel combination of elements, terminal ports, network functions for one port and two port networks, poles and zeros of network functions, significance of poles and zeros, properties of drivingpoint functions and transfer functions, necessary conditions for driving point functions and transfer functions, time domain response from pole-zero plot.

MODULE-IV: THREE PHASE CIRCUITS (10)

Three phase circuits: Star and delta connections, phase sequence, relation between line and phase voltages and currents in balanced systems(both Y&Δ), three phase three wire and three phase four wire systems, analysis of balanced and unbalanced three phase circuits, measurement of active and reactive power.

MODULE-V: FILTERS (07)

Filters: Classification of filters, filter networks, classification of pass band and stop band, characteristic impedance in the pass and stop bands, constant-k low pass filter, high pass filter, m-derived T-section,band pass filter and band elimination filter.

IV. TEXT BOOKS:

1. A Chakrabarthy, “Electric Circuits”, Dhanpat Rai & Sons, 6th Edition, 2010.
2. A Sudhakar, Shyammohan S Palli, “Circuits and Networks”, Tata McGraw Hill, 4th Edition, 2010.

V. REFERENCE BOOKS:

I. John Bird, “Electrical Circuit Theory and technology”, Newnes, 2nd Edition, 2003.
II. C L Wadhwa, “Electrical Circuit Analysis including Passive Network Synthesis”, New Age International, 2nd Edition, 2009.
III. David A Bell, “Electric Circuits”, Oxford University press, 7th Edition, 2009.
IV. M E Van Valkenberg, “Network Analysis”, Prentice Hall India, 3rd Edition, 2014.
V. Rudrapratap, “Getting started with MATLAB: A Quick Introduction for Scientists and Engineers”, Oxford University Press, 1st Edition, 19994.

VI. WEB REFERENCES:

1 https://www.igniteengineers.com
2 https://www.ishuchita.com/PDF/Matlab%20rudrapratap.pdf
3 https://www.ocw.nthu.edu.tw
4 https://www.uotechnology.edu.iq

VII.E-TEXT BOOKS:

1. https://www.bookboon.com/en/concepts-in-electric-circuits-ebook
2. https://www.jntubook.com
3. https://www.allaboutcircuits.com
4. https://www.archive.org

ELECTROMAGNETIC FIELDS

<table>
<tr><td colspan="10">III Semester: EEE</td></tr>
<tr><th>Course Code</th><th>Category</th><th colspan="3">Hours / Week</th><th>Credits</th><th colspan="3">Maximum Marks</th></tr>
<tr><td rowspan="2">AEEC06</td><td rowspan="2">Core</td><td>L</td><td>T</td><td>P</td><td>C</td><td>CIA</td><td>SEE</td><td>Total</td></tr>
<tr><td>3</td><td>1</td><td>0</td><td>4</td><td>30</td><td>70</td><td>100</td></tr>
<tr><td>Contact Classes: 45</td><td>Tutorial Classes: 15</td><td colspan="4">Practical Classes: Nil</td><td colspan="3">Total Classes: 60</td></tr>
<tr><td colspan="10">Prerequisite: Electrical Circuits</td></tr>
</table>

I. COURSE OVERVIEW:

This course will equip the students with good understanding of underlying principles and laws in electromagnetic fields and waves. The concepts of vector algebra, principles and basic laws of electrostatics, characteristics and properties of conductors and dielectrics, behavior of static magnetic field and application of Ampere's law, determination of force in magnetic field and magnetic potential, concept of time varying fields and propagation of electro-magnetic waves.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The behavior of charge under rest with static electric field in terms of electric field intensity, electric displacement and electric potential
II. The charge distribution in conductors, dielectrics and condensers.
III. The sources to develop constant and variable magnetic field to study effect of these fields in terms of magnetic field intensity, magnetic displacement and magnetic potential
IV. The nature of electromagnetic wave propagation in free space, conductors and dielectric materials

III. COURSE SYLLABUS:

MODULE-I: ELECTROSTATICS (10)

Introduction to Cartesian, cylindrical and spherical co-ordinates. Conversion of one type of co-ordinates to another; Electrostatic fields: Coulomb's law, electric field intensity due to line and surface charges, work done in moving a point charge in an electrostatic field, electric potential, properties of potential function, potential gradient, Gauss's law, application of Gauss's law, Maxwell's first law, Laplace's and Poisson's equations, solution of Laplace's equation in one variable.

MODULE-II: CONDUCTORS AND DIELECTRICS (08)

Dipole moment, potential and electric field intensity due to an electric dipole, torque on an electric dipole in an electric field, behavior of conductors in an electric field, electric field inside a dielectric material, polarization, conductor and dielectric, dielectric boundary conditions, capacitance of parallel plate and spherical and coaxial capacitors with composite dielectrics, energy stored and energy density in a static electric field, current density, conduction and convection current densities, Ohm's law in point form, equation of continuity.

MODULE-III: MAGNETOSTATIC (10)

Biot-Savart's law, magnetic field intensity, magnetic field intensity due to a straight current carrying filament, magnetic field intensity due to circular, square and solenoid current carrying wire, relation between magnetic flux, magnetic flux density and magnetic field intensity, Maxwell's second equation, div(B)=0.

Magnetic field intensity due to an infinite sheet of current and a long current carrying filament, point form of Ampere's circuital law, Maxwell's third equation, Curl (H)=Jc, field due to a circular loop, rectangular and square loops

MODULE-IV: FORCE IN MAGNETIC FIELD AND MAGNETIC POTENTIAL (10)

Moving charges in a magnetic field, Lorentz force equation, force on a current element in a magnetic field, force on a straight and a long current carrying conductor in a magnetic field, force between two straight long and parallel current carrying conductors, magnetic dipole and dipole moment, a differential current loop as a magnetic dipole, torque on a current loop placed in a magnetic field;

Vector magnetic potential and its properties, vector magnetic potential due to simple configurations, Poisson's equations, self and mutual inductance, Neumann's formula, determination of self-inductance of a solenoid, toroid and determination of mutual inductance between a straight long wire and a square loop of wire in the same plane, energy stored and density in a magnetic field, characteristics and applications of permanent magnets.

MODULE-V: TIME VARYING FIELDS AND FINITE ELEMENT METHOD (07)

Faraday's laws of electromagnetic induction, integral and point forms, Maxwell's fourth equation, curl (E)=∂B/∂t, statically and dynamically induced EMFs, modification of Maxwell's equations for time varying fields, displacement current.

Derivation of Wave Equation, Uniform Plane Waves, Maxwell's equation in phasor form, Wave equation in Phasor form, Plane waves in free space and in a homogenous material. Wave equation for a conducting medium, Plane waves in loss dielectrics, Propagation in good conductors, Skin effect. Poynting theorem

IV. TEXT BOOKS:

1. K.B. MadhuSahu, " Eelectromagnetic Fields", Scitech Ltd., 2nd Edition, 2014.
2. David J Griffiths, "Introduction to Electrodynamics" Pearson Education Ltd., 4th Edition, 2014.
3. Sunil Bhooshan, "Fundamentals of Engineering Electromagnetics", Oxford University Press, 1st Edition, 2012.
4. E Kuffel, W S Zaengl, J Kuffel, "High Voltage Engineering Fundamentals", Newnes, 2nd Edition, 2000.

V. REFERENCE BOOKS:

1. Matthew N O Sadiku, S V Kulkarni, "Principles of Electromagnetics", Oxford University Press, 6th Edition, 2015.
2. AS Mahajan , AA Rangwala "Electricity And Magnetism", McGraw Hill Publications, 1st Edition, 2000.
3. MS Naidu, V Kamaraju "High Voltage Engineering", McGraw Hill Publications, 3rd Edition, 2013.
4. William H Hayt, John A Buck, "Problems and Solutions in Electromagnetics", McGraw Hill Publications, 1st Edition, 2010.

VI. WEB REFERENCES:

1. https://www.calvin.edu/~pribeiro/courses/engr315/EMFT_Book.pdf
2. https://www.web.mit.edu/viz/EM/visualizations/coursenotes/modules/guide02.pdf
3. https://www.nptel.ac.in/courses/108106073/
4. https://www.iare.ac.in

VII. E-TEXT BOOKS:

1. https://www.bookboon.com/en/electromagnetism-for-electronic-engineers
2. https://www.books.google.co.in/books/.../Fundamentals of Electromagnetic Fields
3. https://www.aliexpress.com/item/EBOOK...Electromagnetic-Fields-2

ANALOG ELECTRONICS

III Semester: EEE								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
AECC07	Core	L	T	P	C	CIA	SEE	Total
		3	0	0	3	30	70	100
Contact Classes: 45	Tutorial Classes: Nil	Practical Classes: Nil				Total Classes: 45		
Prerequisite: : Electrical Circuits , Linear Algebra and Calculus								

I. COURSE OVERVIEW:

This course introduces the fundamentals and principles of semiconductor devices and circuits. Apply the characteristics of diodes, bipolar, uni-polar transistors and operational amplifiers for designing rectifiers, clippers, clampers, amplifiers, oscillators and waveform generators. It provides skills for analyzing amplifier circuits using small signal model and hybrid pi model.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The operational principles of analog electronic circuits such as feedback amplifiers and operational amplifiers

II. The analog circuits fundamental theory to build signal conversion circuits, filter circuits, Data converters and Automatic Gain Control.

III. The analog circuits applications in the advanced fields power electronics such as power factor monitoring circuits, power quality measurement, SMPS and battery controls

III. COURSE SYLLABUS:

MODULE-I: DIODE CIRCUITS (10)

P-N junction diode, I-V characteristics of a diode; review of half-wave and full-wave rectifiers, clamping and clipping circuits. Input output characteristics of BJT in CB, CE, CC configurations, biasing circuits, Load line analysis, common emitter, common base and common collector amplifiers; Small signal equivalent circuits.

MODULE-II: MOSFET CIRCUITS (08)

MOSFET structure and I-V characteristics. MOSFET as a switch. small signal equivalent circuits - gain, input and output impedances, small-signal model and common-source, common-gate and common-drain amplifiers, trans conductance, high frequency equivalent circuit.

MODULE-III: MULTI-STAGE AND POWER AMPLIFIERS (10)

Classification of Amplifiers, Distortion in amplifiers, Different coupling schemes used in amplifiers, Frequency response and Analysis of multistage amplifiers, Cascade amplifier, Darlington pair.

Transistor at High Frequency: Hybrid - model of Common Emitter transistor model, f_α, β and unity gain bandwidth, Gain band width product. Differential Amplifiers, Power amplifiers - Class A, Class B, Class C, Class AB.

MODULE-IV: FEEDBACK AMPLIFIERS (10)

Concepts of feedback: Classification of feedback amplifiers, general characteristics of Negative feedback amplifiers, effect of feedback on amplifier characteristics, voltage series, voltage shunt, current series and current shunt feedback configurations, simple problems; Oscillators: Condition for Oscillations, RC type Oscillators RC phase shift and Wien-bridge Oscillators, LC type Oscillators, generalized analysis of LC Oscillators, Hartley and Colpitts oscillators;

MODULE-V: OPERATIONAL AMPLIFIERS (07)

Ideal op-amp, Output offset voltage, input bias current, input offset current, slew rate, gain bandwidth product, Inverting and non-inverting amplifier, Differentiator, integrator, Square-wave and triangular-wave generators.

IV. TEXT BOOKS:

1. Jacob Millman, Christos C Halkias, “Integrated Electronics”, McGraw Hill Education, 2^{nd} Edition 2010.
2. Ramakanth A, Gayakwad, “Op-Amps & Linear Ics”, PHI, 2003.

V. REFERENCE BOOKS:

1. Thomas L. Floyd, “Electronic Devices Conventional and Current Version”, Pearson, 2013.
2. J. Millman and A. Grabel, “Microelectronics”, McGraw Hill Education, 1988.
3. P. Horowitz and W. Hill, “The Art of Electronics”, Cambridge University Press, 1989.
4. P. R. Gray, R. G. Meyer and S. Lewis, “Analysis and Design of Analog Integrated Circuits”, John Wiley & Sons, 2001

VI. WEB REFERENCES:

1. http://www-mdp.eng.cam.ac.uk/web/library/enginfo/electrical/hong1.pdf
2. https://archive.org/details/ElectronicDevicesCircuits
3. http://nptel.ac.in/courses/Webcourse-contents/IIT-ROORKEE/BASIC ELECTRONICS/home_page.htm
4. www.nptel.ac.in
5. notes.specworld.in/pdc-pulse-and-digital-circuits

VII. E-TEXT BOOKS:

1. http://services.eng.uts.edu.au/pmcl/ec/Downloads/LectureNotes.pdf
2. http://nptel.ac.in/courses/122106025/
3. http://www.freebookcentre.net/electronics-ebooks-download/Electronic-Devices-and-Circuits-(PDF-313p).html
4. http:// www.introni.it/pdf/Millman-Taub- Pulse and Digital Switching Waveforms 1965.pdf
5. https://www.jntubook.com/pulse-digital-circuits-textbook-free-download/

DC MACHINES AND TRANSFORMERS

<table>
<tr><td colspan="10">III Semester: EEE</td></tr>
<tr><th>Course Code</th><th>Category</th><th colspan="3">Hours / Week</th><th>Credits</th><th colspan="3">Maximum Marks</th></tr>
<tr><td rowspan="2">AEEC07</td><td rowspan="2">Core</td><td>L</td><td>T</td><td>P</td><td>C</td><td>CIA</td><td>SEE</td><td>Total</td></tr>
<tr><td>3</td><td>0</td><td>0</td><td>3</td><td>30</td><td>70</td><td>100</td></tr>
<tr><td>Contact Classes: 45</td><td>Tutorial Classes: Nil</td><td colspan="4">Practical Classes: Nil</td><td colspan="3">Total Classes: 45</td></tr>
<tr><td colspan="10">Prerequisite: Electrical Circuits (AEEC02). Engineering Physics (AHSCO3)</td></tr>
</table>

I. COURSE OVERVIEW:

This course deals with the basic theory, construction, operation, performance characteristics and application of electromechanical energy conversion devices such as DC generators and motors. It also gives an in-depth knowledge on the operation of single phase and three phase transformers and it's testing. It also focus on the auto transformers, on-load, off-load tap changers which are widely used in real time applications.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The principles of single excited and multiple excited systems leading to the energy balance equations.
II. The construction, working and operation of self and separately excited DC machines.
III. The performance characteristics of different DC machines when they are under no load and load conditions.
IV. The energy transformation using single and poly phase transformers under no load and load conditions.

III. COURSE SYLLABUS:

MODULE-I: DC GENERATORS (10)

DC generators: Principle of operation, construction, lap and wave windings, simplex and multiplex windings, commutator, EMF equation, types of DC generators, Armature reaction: Cross magnetization and demagnetization, ampere turns per pole, compensating winding; Commutation: Methods of improving commutation; Open circuit characteristics, voltage buildup, critical field resistance and critical speed, causes for failure to self-excite and remedial measures, load characteristics of shunt, series and compound generators; Conditions and necessity for parallel operation, load sharing, equalizer bars, cross connection of field windings, numerical problems.

MODULE-II: DC MOTORS (08)

DC motors: Principle of operation, back EMF, torque equation, types of DC motors, condition for maximum power developed, armature reaction and commutation, characteristics, types of starters, numerical problems.

MODULE-III: PERFORMANCE OF DC MACHINES (10)

Losses and efficiency: Types of losses, efficiency, condition for maximum efficiency

Speed Control Methods: Speed control of DC machines; Testing methods: Swinburne's test, brake test, retardation test, separation of stray losses, Hopkinson's test, and field's test, numerical problems

MODULE-IV: SINGLE PHASE TRANSFORMERS (10)

Single phase transformers: Principle of operation, construction, types of transformers, EMF equation, concept of leakage flux and leakage reactance, operation of transformer under no-load and on-load, phasor diagrams, equivalent circuit, efficiency, regulation and all day efficiency; Cooling methods; Testing of transformers: objectives, polarity test, measurement of resistance, OC and SC tests, back to back test, heat run test, parallel operation, numerical problems.

MODULE-V: POLY PHASE TRANSFORMERS (07)

Three phase transformer: Principle of operation, star to star, delta to delta, star to delta, delta to star, three phase to six phase, open delta connection, Scott connection; Auto transformers: Principles of operation, equivalent circuit, merits and demerits, no load and on load tap changers, harmonic reduction in phase voltages, numerical problems.

IV. TEXT BOOKS:

1. P S Bimbhra, "Electrical Machinery", Khanna Publishers, 1^{st} Edition, 2011.
2. I J Nagrath and D P Kothari, "Electric Machines", McGraw Hill Education, 1^{st} Edition, 2010.
3. J B Guptha "Theory and performance of Electrical machines", S.K.Kataria & Sons Publishers 14^{th} Edition, 2009.

V. REFERENCE BOOKS:

1. M G Say, E O Taylor, "Direct Current Machines", Longman Higher Education, 1^{st} Edition, 1985.
2. M G Say, "Performance and design of AC machines", CBS Publishers, 1^{st} Edition, 2002.
3. A E Fitzgerald and C Kingsley, "Electric Machinery", New York, McGraw Hill Education, 1^{st} Edition, 2013.
4. M V Deshpande, "Electrical Machines", PHI Learning Private Limited, 3^{rd} Edition, 2011.

VI. WEB REFERENCES:

1. https://www.electrical4u.com
2. https://www.freevideolectures.com

VII. E-TEXT BOOKS:

1. https://www.freeengineeringbooks.com
2. https://www.pdfdrive.com/textbook-of-electrical-technology-ac-and-dc-machines-d184089760.html

DATA STRUCTURES

III Semester: Common for all branches								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
ACSC08	**Core**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		3	0	0	3	30	70	100
Contact Classes: 45	**Tutorial Classes: Nil**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisites: Python Programming								

I. COURSE OVERVIEW:

The course covers some of the general-purpose data structures and algorithms, and software development. Topics covered include managing complexity, analysis, static data structures, dynamic data structures and hashing mechanisms. The main objective of the course is to teach the students how to select and design data structures and algorithms that are appropriate for problems that they might encounter in real life. This course reaches to student by power point presentations, lecture notes, and lab which involve the problem solving in mathematical and engineering areas.

II. COURSE OBJECTIVES:

The students will try to learn:

I. To provide students with skills needed to understand and analyze performance trade-offs of different algorithms / implementations and asymptotic analysis of their running time and memory usage.
II. To provide knowledge of basic abstract data types (ADT) and associated algorithms: stacks, queues, lists, tree, graphs, hashing and sorting, selection and searching.
III. The fundamentals of how to store, retrieve, and process data efficiently.
IV. To provide practice by specifying and implementing these data structures and algorithms in Python.
V. Understand essential for future programming and software engineering courses.

III. SYLLABUS:

MODULE – I: INTRODUCTION TO DATA STRUCTURES, SEARCHING AND SORTING (09)

Basic concepts: Introduction to data structures, classification of data structures, operations on data structures; Algorithm Specification, Recursive algorithms, Data Abstraction, Performance analysis- time complexity and space complexity, Asymptotic Notation-Big O, Omega, and Theta notations. Introduction to Linear and Non Linear data structures, Searching techniques: Linear and Binary search; Sorting techniques: Bubble, Selection, Insertion, Quick and Merge Sort and comparison of sorting algorithms.

MODULE – II: LINEAR DATA STRUCTURES (09)

Stacks: Stack ADT, definition and operations, Implementations of stacks using array, applications of stacks, Arithmetic expression conversion and evaluation; Queues: Primitive operations; Implementation of queues using Arrays, applications of linear queue, circular queue and double ended queue (deque).

MODULE – III: LINKED LISTS (09)

Linked lists: Introduction, singly linked list, representation of a linked list in memory, operations on a single linked list; Applications of linked lists: Polynomial representation and sparse matrix manipulation.

Types of linked lists: Circular linked lists, doubly linked lists; Linked list representation and operations of Stack, linked list representation and operations of queue.

MODULE - IV NON LINEAR DATA STRUCTURES (09)

Trees: Basic concept, binary tree, binary tree representation, array and linked representations, binary tree traversal, binary tree variants, threaded binary trees, application of trees, Graphs: Basic concept, graph terminology, Graph Representations - Adjacency matrix, Adjacency lists, graph implementation, Graph traversals – BFS, DFS, Application of graphs, Minimum spanning trees – Prims and Kruskal algorithms.

MODULE - V BINARY TREES AND HASHING (09)

Binary search trees: Binary search trees, properties and operations; Balanced search trees: AVL trees; Introduction to M-Way search trees, B trees; Hashing and collision: Introduction, hash tables, hash functions, collisions, applications of hashing.

IV. TEXT BOOKS:

1. Rance D. Necaise, “Data Structures and Algorithms using Python”, Wiley Student Edition.
2. Benjamin Baka, David Julian, “Python Data Structures and Algorithms”, Packt Publishers, 2017.

V. REFERENCE BOOKS:

1. S. Lipschutz, “Data Structures”, Tata McGraw Hill Education, 1st Edition, 2008.
2. D. Samanta, “Classic Data Structures”, PHI Learning, 2nd Edition, 2004.

VI. WEB REFERENCES:

. https://www.tutorialspoint.com/data_structures_algorithms/algorithms_basics.htm
. https://www.codechef.com/certification/data-structures-and-algorithms/prepare
. https://www.cs.auckland.ac.nz/software/AlgAnim/dsToC.html
. https://online-learning.harvard.edu/course/data-structures-and-algorithms

XPERIENTIAL ENGINEERING EDUCATION (EXEED) – PROTOTYPE / DESIGN BUILDING

III Semester: Common for all branches								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
ACSC09	Foundation	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		2	0	0	1	30	70	100
Contact Classes: 28	**Tutorial Classes: Nil**	**Practical Classes: Nil**				**Total Classes: 28**		
Prerequisite: There are no prerequisites to take this course								

I. COURSE OVERVIEW:

This course provides an overall exposure to the various methods and tools of prototyping. This course discusses Low- Fidelity, paper, wireframing and tool based prototyping techniques along with design principles and patterns.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The basic principles and design aspect of prototyping.
II. The various techniques, design guidelines and patterns.
III. The applications of prototyping using various tools and platforms.

WEEK NO	TOPIC
WEEK – I	An introduction to Prototyping
WEEK – II	Low - Fidelity Prototyping and Paper Prototyping
WEEK – III	Wireframing and Tool based Prototyping
WEEK – IV	Physical Low- Fidelity Prototyping
WEEK – V	Tool based prototyping
WEEK – VI	Design Principles and Patterns- Graphic Design
WEEK – VII	Design Principles and Patterns- Interaction Design
WEEK –VIII	Commercial design guidelines and standards.
WEEK - IX	Universal design: Sensory and cognitive impairments
WEEK - X	Universal design: Tools, Limitations and standards
WEEK - XI	Introduction platforms and context : Mobile UI design, Wearable
WEEK - XII	Introduction platforms and context : Automotive user interface
WEEK - XIII	Introduction platforms and context : IoT and Physical Computing
WEEK - XIV	Assessment

NETWORK ANALYSIS AND SCIENTIFIC COMPUTING LABORATORY

III Semester: EEE								
Course Code	Category	Hours / Week			Credits	Maximum Marks		
AEEC08	Core	L	T	P	C	CIA	SEE	Total
		1	0	2	2	30	70	100
Contact Classes: Nil	Tutorial Classes: Nil	Practical Classes: 36				Total Classes:36		
Prerequisite: Electrical Circuits, Linear Algebra and Calculus								

I. COURSE OVERVIEW:

The Network Analysis and Scientific Computing Laboratory is designed to give hands-on experience on virtual instrumentation through digital simulation techniques. These techniques enable the students in examining characteristics of DC and AC circuits, filters, solution of differential equation, generation of three phase and complex wave forms using MATLAB.

II. COURSE OBJECTIVES:

The students will try to learn:

I. Time varying characteristics of series and parallel circuits using MATLAB.
II. Transfer function of electrical circuits using MATLAB.
III. Relations between electrical quantities in complex electrical networks using MATLAB.
IV. The performance of single phase and three phase circuits using Lab View.

III. COURSE OUTCOMES:

CO1	Identify the symbols, tool kits and connections in Simulink environment for computing the quantities associated with electrical circuits
CO2	Examine the transfer function for studying transient response of RL, RC and RLC circuits.
CO3	Analyze the virtual instrumentation (VI) using control loops, arrays, charts and graphs.
CO4	Determine various alternating quantities of single phase and three phase signals generated in MATLAB/ LabVIEW.
CO5	Design the various sensors for measuring electrical and non-electrical quantities through digital simulation.

IV. SYLLABUS:

EXERCISES ON NETWORK ANALYSIS AND SCIENTIFIC COMPUTING LABORATORY

Note: Students are encouraged to bring their own laptops for laboratory practice sessions.

1. Getting Started Exercises

Introduction to MATLAB

Identify the symbols, tool kits and connections in Simulink environment for computing the quantities associated with electrical circuits.

MATLAB is a powerful software environment widely used in various fields, including electrical engineering and circuit analysis. It provides a range of symbols, toolkits, and connections related to electrical circuits to assist engineers and researchers in designing, analyzing, and simulating electrical circuits. Here's an introduction to some of the key features and tools MATLAB offers in this regard:

1. Symbolic Math Toolbox: MATLAB's Symbolic Math Toolbox allows you to perform symbolic computations, which are crucial for electrical circuit analysis. You can define symbolic variables for circuit components like resistors, capacitors, and inductors and manipulate algebraic equations symbolically. This is useful for solving circuit equations and obtaining transfer functions.
2. Simulink: Simulink is a graphical environment within MATLAB for modeling, simulating, and analyzing dynamic systems, including electrical circuits. It provides a vast library of pre-built electrical components and blocks, making it easier to create circuit models and simulate their behavior.
3. Control System Toolbox: This toolbox contains tools and functions for analyzing and designing control systems, which are essential for understanding the behavior of electrical circuits. You can use it to design controllers, analyze stability, and simulate the response of electrical systems.
4. Electrical Circuit Analysis Toolbox: There are also third-party toolboxes available for MATLAB, such as the Electrical Circuit Analysis Toolbox, that provide specialized tools for electrical circuit analysis. These toolboxes often include features like circuit simulation, AC/DC analysis, and more.
5. SimPowerSystems (formerly known as SimElectronics): SimPowerSystems is a specialized toolbox in Simulink for modeling and simulating electrical circuits and systems. It includes a wide range of electrical components, such as transformers, generators, and motors, allowing you to create detailed electrical system models.
6. Simscape Electrical: Simscape Electrical is another Simulink toolbox that focuses on modeling and simulating electrical power systems. It includes components like power electronics devices, electrical machines, and power generation and distribution elements.
7. Signal Processing Toolbox: This toolbox is valuable for analyzing signals within electrical circuits. You can use it for tasks like filtering noisy signals, performing Fourier analysis, and extracting important information from sensor data.
8. Data Acquisition Toolbox: If you're working with hardware experiments or data acquisition systems, this toolbox can help you connect MATLAB to external hardware, making it easier to acquire and analyze data from real-world electrical circuits.

9. Instrument Control Toolbox: This toolbox enables you to communicate with and control external instruments like oscilloscopes, signal generators, and mustimeters, which is beneficial for experimental work in electrical engineering.
10. Custom Functions and Scripts: MATLAB allows you to create custom functions and scripts to model and analyze specific aspects of electrical circuits. You can define your own functions for circuit equations, transfer functions, and more.
    Creating syntax programs for electrical circuit analysis in MATLAB often involves using built-in functions, toolboxes, and custom code to define circuits, simulate their behavior, and analyze results. Below are some examples of syntax programs for common tasks in electrical circuit analysis using MATLAB:

Hints

1. Defining Circuit Components and Equations:

```
 % Define symbolic variables
syms V1 V2 R1 R2 C1 L1 s

% Define resistor values
R1 = 100; % Ohms
R2 = 200; % Ohms

% Define capacitor and inductor values
C1 = 0.1; % Farads
L1 = 0.5; % Henrys

% Define Kirchhoff's laws for an RC circuit
Eq1 = V1 - R1 * I1 - V2 == 0;
Eq2 = V2 - R2 * I2 - I1/C1 == 0;

% Define Laplace domain equations
Eq1_Laplace = laplace (Eq1, t, s);
Eq2_Laplace = laplace (Eq2, t, s);
```

2. Solving Circuit Equations:

```
% Solve the Laplace domain equations symbolically
I1_Laplace = solve (Eq1_Laplace, I1);
I2_Laplace = solve (Eq2_Laplace, I2);

% Inverse Laplace transform to get time-domain solutions
I1_time = ilaplace (I1_Laplace, s, t);
I2_time = ilaplace (I2_Laplace, s, t);
```

3. Simulating Circuits in Simulink:

```
% Create a Simulink model for an RLC circuit
model = 'RLC_Circuit';
open_system(new_system(model));

% Add components and connections using Simulink blocks
add_block ('Simulink/Continuous/Resistor', [model '/R1']);
```

```
add_block ('Simulink/Continuous/Inductor', [model '/L1']);
add_block ('Simulink/Continuous/Capacitor', [model '/C1']);
add_block ('Simulink/Sources/Step', [model '/Step Voltage']);
add_line (model, 'Step Voltage/1', 'R1/1');
% Add connections as needed

% Set component values and simulation parameters
set_param ([model '/R1'], 'Resistance', '100');
set_param ([model '/L1'], 'Inductance', '0.5');
set_param ([model '/C1'], 'Capacitance', '0.1');
set_param (model, 'Stop Time', '5');

% Run the simulation
sim(model);
```

4. Analyzing Frequency Response:

```
% Define a transfer function for an RC circuit
num = [1];
den = [R1*C1 1];
sys = tf (num, den);

% Bode plot for frequency response
bode(sys);
```

5. Data Analysis and Plotting:

```
% Import experimental data from a file
data = import data('experimental_data.csv');
time = data (: 1);
voltage = data (: 2);

% Plot voltage-time graph
plot (time, voltage);
xlabel ('Time (s)');
ylabel ('Voltage (V)');
title ('Voltage vs. Time');
```

Try:

a. Circuit Equations:

i. Write down the Kirchhoff's Voltage Law (KVL) equation for a series RC circuit involving a resistor (R) and a capacitor (C). Assume a constant voltage source V_S.

ii. Write down the Kirchhoff's Voltage Law (KVL) equation for a series RL circuit involving a resistor (R) and an inductor (L). Assume a time-varying current source I_S(t).

b. MATLAB Code: Write MATLAB code to:

i. Solve the KVL equation for the RC circuit (from question 2a) symbolically for V_C as a function of time, assuming initial conditions.

ii. Create a Simulink model for the RL circuit (from question 2b) with appropriate components and connections. Set the simulation parameters and run a transient analysis for a specified time duration.

Hint: Save your MATLAB code as a separate script file (.m) for questions

c. Transfer Function:

i. Define the transfer function H(s) for the RC circuit in terms of Laplace variables.

ii. Plot the Bode plot of H(s) to analyze the frequency response of the RC circuit.

d. Data Analysis:

i. Import experimental voltage and time data from a CSV file into MATLAB.

ii. Plot the voltage-time graph and label the axes appropriately.

e. Circuit Toolbox:

Mention at least two MATLAB toolboxes that are useful for electrical circuit analysis. Explain briefly how each toolbox can assist in circuit analysis.

f. If R = 10 Ohms and the current is increased from 0 to 10 A with increments of 2A, write a MATLAB program to generate a table of current, voltage and power dissipation.

2. Transient Response of Series RL, RC and RLC Circuits:

Examine the time varying characteristics of series RL, RC and RLC circuits for given values of R, L and C using MATLAB software.

2.1 Transient Response of Series RL

In this MATLAB program:

i. Circuit parameters (resistance, inductance, and applied voltage) are defined.
ii. The time constant (τ) is calculated as L / R.
iii. A symbolic variable **t** is defined for time.
iv. The differential equation representing the transient response of the RL circuit is defined using **diff** and ==.
v. The **dsolve** function is used to solve the differential equation symbolically.
vi. A time vector (**'t_span'**) is created for the simulation.
vii. IL(t) is numerically evaluated for the given time vector.
viii. The transient response is plotted.

Hints

```
% Define circuit parameters
R = 100; % Resistance in ohms
L = 0.5; % Inductance in henrys
Vin = 10; % Applied voltage in volts

% Calculate the time constant τ
tau = L / R;

% Define the symbolic variable for time
syms t

% Define the differential equation for the current IL(t)
IL = sym('IL(t)');
eqn = L * diff(IL, t) + R * IL == Vin;

% Solve the differential equation
IL_solution = dsolve (eqn, IL);
IL_solution = simplify(IL_solution);

% Create a time vector and evaluate IL(t
```

```
time = 0:0.01:2*tau; % Adjust the time span as needed
IL_numeric = double (subs (IL_solution, t, time));

% Plot the transient response
plot (time, IL_numeric);
xlabel ('Time (s)');
ylabel ('Current (A)');
title ('Transient Response of RL Circuit');
grid on;
```

Try:

a. Circuit Parameters:

Given the following values:

i. Resistor (R): 150 ohms
ii. Inductor (L): 0.3 H
iii. Applied Voltage (V_in): 12 V

Calculate the time constant (τ) of the RL circuit.

b. Transient Analysis:

iv. Write down the first-order differential equation governing the transient response of the RL circuit in terms of the current (I_L) through the inductor.
v. Solve the differential equation symbolically to obtain the expression for I_L(t) as a function of time (t).

c. MATLAB Simulation: Write MATLAB code to:

i. Define the circuit parameters (R, L, and V_in).
ii. Simulate the transient response of the RL circuit using the 'l_{sim}' function. Use a time vector from 0 to 5τ with a suitable time step.
iii. Plot the transient response of the current (I_L) as a function of time. Label the axes appropriately.

d. Time Constant Verification:

i. Calculate the time constant τ numerically from the transient response obtained in question 3c.
ii. Compare the numerically calculated time constant with the value obtained in question 1. Explain any differences or similarities.

e. Analysis:

Based on the transient response plot obtained in question iii, discuss the behavior of the current in the RL circuit as it responds to the voltage step input. Specifically, explain the time constant, the initial current, and the behavior as time progresses.

2.2 Transient Response of Series RC

In this MATLAB code:

i. Circuit parameters such as resistance (R), capacitance (C), and applied voltage (Vin) are defined.
ii. The time constant (τ) is calculated as R * C.
iii. A symbolic variable t is defined for time.

iv. The differential equation representing the transient response of the RC circuit is defined using diff and ==.
v. The dsolve function is used to solve the differential equation symbolically.
vi. A time vector (t_span) is created for the simulation.
vii. VC(t) is numerically evaluated for the given time vector.
viii. The transient response is plotted, showing how the voltage across the capacitor changes over time in response to a step input voltage.

You can adjust the time span in t_span to observe the transient response for the desired duration.

Hints

```
% Define circuit parameters
R = 100; % Resistance in ohms
C = 0.1; % Capacitance in farads
Vin = 5; % Applied voltage in volts

% Calculate the time constant τ
tau = R * C;

% Define the symbolic variable for time
syms t

% Define the differential equation for the voltage VC(t) across the capacitor
VC = sym('VC(t)');
eqn = R * diff (VC, t) + VC == Vin;

% Solve the differential equation symbolically
VC_solution = dsolve (eqn, VC);
VC_solution = simplify(VC_solution);

% Create a time vector
t_span = 0:0.01:5*tau; % Adjust the time span as needed

% Evaluate VC(t) for the given time vector
VC_numeric = double (subs (VC_solution, t, t_span));

% Plot the transient response
plot (t_span, VC_numeric);
xlabel ('Time (s)');
ylabel ('Voltage across Capacitor (V)');
title ('Transient Response of RC Circuit');
grid on;
```

Try:

a. Circuit Parameters:

Given the following values:

i. Resistor (R): 220 ohms
ii. Inductor (L): 0.01 F
iii. Applied Voltage (V_in): 10 V

Calculate the time constant (τ) of the RC circuit.

b. Transient Analysis:

i. Write down the first-order differential equation governing the transient response of the RC circuit in terms of the voltage (V_C) across the capacitor.
ii. Solve the differential equation symbolically to obtain the expression for V_C(t) as a function of time (t).

c. MATLAB Simulation: Write MATLAB code to:

i. Define the circuit parameters (R, C, and V_in).
ii Simulate the transient response of the RC circuit using the 'lsim' function. Use a time vector from 0 to 5τ with a suitable time step.
iii. Plot the transient response of the voltage across the capacitor (V_C) as a function of time. Label the axes appropriately.

d. Time Constant Verification:

i. Calculate the time constant τ numerically from the transient response obtained in question 3c.
ii. Compare the numerically calculated time constant with the value obtained in question 1. Explain any differences or similarities.

e. Analysis:

Based on the transient response plot obtained in question c(iii), discuss the behavior of the current in the RL circuit as it responds to the voltage step input. Specifically, explain the time constant, the initial current, and the behavior as time progresses.

2.3 Transient Response of Series RLC

In this MATLAB code:

I. Circuit parameters such as resistance (R), inductance (L), capacitance (C), and applied voltage (Vin) are defined.
II. The natural frequency (wn) and damping ratio (zeta) are calculated based on R, L, and C.
III. The time constant (τ) is calculated as 1 / (wn * sqrt (1 - zeta^2)).
IV. A symbolic variable **'t'** is defined for time.
V. The differential equation representing the transient response of the RLC circuit is defined using **'diff'** and ==.
VI. The **dsolve** function is used to solve the differential equation symbolically.
VII. A time vector (**'t_span'**) is created for the simulation.
VIII. VC(t) is numerically evaluated for the given time vector.
IX. The transient response is plotted, showing how the voltage across the capacitor changes over time in response to the applied voltage.

You can adjust the time span in t_span as needed to observe the transient response for the desired duration.

Hints

```
% Define circuit parameters
R = 100; % Resistance in ohms
L = 0.2; % Inductance in henrys
```

```
C = 0.01; % Capacitance in farads
Vin = 10; % Applied voltage in volts

% Calculate the natural frequency (wn) and damping ratio (zeta)
wn = 1 / sqrt(L * C);
zeta = R / (2 * L * wn);

% Calculate the time constant (τ)
tau = 1 / (wn * (1 - zeta^2) ^0.5);

% Define the symbolic variable for time
syms t

% Define the differential equation for the voltage VC(t) across the capacitor
VC = sym('VC(t)');
eqn = L * diff (VC, t, 2) + R * diff (VC, t) + (1/C) * VC == Vin;

% Solve the differential equation symbolically
VC_solution = dsolve (eqn, VC);
VC_solution = simplify(VC_solution);

% Create a time vector
t_span = 0:0.001:5*tau; % Adjust the time span as needed

% Evaluate VC(t) for the given time vector
VC_numeric = double (subs (VC_solution, t, t_span));

% Plot the transient response
plot (t_span, VC_numeric);
xlabel ('Time (s)');
ylabel ('Voltage across Capacitor (V)');
title ('Transient Response of RLC Circuit');
grid on;
```

Try

a. Circuit Parameters:

Given the following values:

I. Resistor (R): 220 ohms
ii. Inductor (L): 0.1 H
iii. Capacitor (C): 0.01 F
iv. Applied Voltage (V_in): 12 V

Calculate the natural frequency (wn), damping ratio (zeta), and time constant (τ) of the RLC circuit.

b. Transient Analysis:

i. Write down the second-order differential equation governing the transient response of the
ii. RLC circuit in terms of the voltage (V_C) across the capacitor.
iii. Solve the differential equation symbolically to obtain the expression for V_C(t) as a function of time (t).

c. MATLAB Simulation: Write MATLAB code to:

i. Define the circuit parameters (R, L, C, and V_in).

ii. Simulate the transient response of the RLC circuit using the lsim function. Use a time vector from 0 to 5τ with a suitable time step.

iii. Plot the transient response of the voltage across the capacitor (V_C) as a function of time. Label the axes appropriately.

d. Time Constant Verification:

i. Calculate the natural frequency (wn), damping ratio (zeta), and time constant (τ) numerically from the transient response obtained in question 3c.

ii. Compare the numerically calculated values with the values obtained in question 1. Explain any differences or similarities.

3. Solving Differential Equations

Perform the solution of differential equation which is representing mathematical model of electric circuit using MATLAB software.

Define the Differential Equation: First, define the differential equation that represents the circuit. This equation depends on the specific circuit components and the mathematical model you're using. For instance, consider an RC circuit with a resistor (R) and a capacitor (C) connected in series. The differential equation for the voltage across the capacitor (Vc) can be written as:

$$RC\frac{dV_c}{dt} + V_c - V_{in}$$

where RC is the time constant, $\frac{dV_c}{dt}$ represents the derivative of Vc with respect to time, and Vin is the applied voltage.

Convert to Symbolic Variables: In MATLAB, convert the equation into symbolic form using the **'syms'** function and define the symbolic variables. For example:

Hints

```
syms Vc(t) Vin R C
eqn = R * C * diff(Vc(t), t) + Vc(t) == Vin;
```

Solve the Differential Equation: Use the '**dsolve'** function to solve the differential equation symbolically:

```
Vc_solution = dsolve(eqn);
```

Define Initial Conditions: If your circuit has initial conditions (e.g., Vc(0) = 0), you can specify them in the '**dsolve'** function to get a more specific solution.

```
Vc_solution = dsolve(eqn, Vc(0) == 0); % Example initial condition
```

Plot the Solution: Create a time vector and evaluate the symbolic solution for Vc(t) for the given time vector. Then, plot the transient response:

```
t = 0:0.01:5*RC; % Adjust the time span as needed
Vc_numeric = double (subs (Vc_solution, t));
plot (t, Vc_numeric);
xlabel ('Time (s)');
```

```
ylabel ('Voltage across Capacitor (V)');
title ('Transient Response of RC Circuit');
```

Try

I. Write down the differential equation representing the voltage across the capacitor (Vc) in the RC circuit. Include all the relevant terms and variables.

II. Define the symbolic variables and the differential equation in MATLAB to represent the RC circuit.

III. Use the **dsolve** function to solve the differential equation symbolically for $V_c(t)$ Assume an initial condition $V_c(0) = 0$

IV. Write MATLAB code to:

Define the circuit parameters:

Resistance R: 100 ohms

Capacitance C: 0.01 F

Applied Voltage V_{in}: 10 V

V. Simulate and plot the transient response of the voltage across the capacitor $V_c(t)$ for a time span of 0 to 5 times the time constant (τ).

VI. Label the axes appropriately and provide a title for the plot.

VII. Calculate the time constant (τ) of the RC circuit using the given values of R and C

VIII. Based on the transient response plot obtained in question v, discuss the behavior of the voltage across the capacitor (Vc) as it responds to the voltage step input. Explain the concept of the time constant (τ) and its significance in the transient response.

4. Transfer Function of Electrical Circuit

Determine the voltage transfer function of series RLC electrical circuit for frequencies from 10Hz to 100KHz using MATLAB software.

To determine the voltage, transfer function of a series RLC electrical circuit for a range of frequencies from 10 Hz to 100 kHz using MATLAB, you can follow these steps

Define Circuit Parameters:

- Resistance (R) in ohms
- Inductance (L) in henrys
- Capacitance (C) in farads

Define Symbolic Variables:

- s as the complex frequency variable

Create the Transfer Function:

The transfer function (H(s)) for a series RLC circuit can be defined as follows:

$$H(s) = \frac{V_{out}(s)}{V_{in}(s)} - \frac{1}{LC^2+RCs+1}$$

Where:

$V_{out}(s)$ is the Laplace-transformed output voltage?

$V_{in}(s)$ is the Laplace-transformed input voltage?

L is the inductance.
C is the capacitance.
R is the resistance.
s is the complex frequency variable.

Calculate and Plot the Frequency Response:

HINTS

```
% Define circuit parameters
R = 100; % Resistance in ohms
L = 0.1; % Inductance in henrys
C = 0.01e-6; % Capacitance in farads (1 uF)
frequencies = logspace (1, 5, 100); % Frequency range from 10 Hz to 100 kHz

% Define the complex frequency variable s
s = j*2*pi*frequencies;

% Calculate the transfer function H(s)
H = 1. / (L*C*s.^2 + R*C*s + 1);

% Calculate magnitude and phase of H(s)
magnitude_H = abs(H);
phase_H = angle(H);

% Plot the Bode plot (magnitude and phase)
subplot (2,1,1);
semilogx (frequencies, 20*log10(magnitude_H));
title ('Bode Plot of RLC Circuit');
xlabel ('Frequency (Hz)');
ylabel ('Magnitude (dB)');
grid on;

subplot (2,1,2);
semilogx (frequencies, rad2deg(phase_H));
xlabel ('Frequency (Hz)');
ylabel ('Phase (degrees)');
grid on;
```

In this code:

- We define the circuit parameters (R, L, and C).
- We create a range of frequencies from 10 Hz to 100 kHz using **'logspace'**.
- We define the complex frequency variable s as $j\omega$, where ω is the angular frequency.
- The transfer function $H(s)$ is calculated for each frequency in the range.
- We calculate the magnitude and phase of $H(s)$
- Finally, we plot the Bode plot, showing the magnitude and phase response of the RLC circuit as a function of frequency.

Try

I. a series RLC electrical circuit with the following parameters:

Resistance (R): 150 ohms

Inductance (L): 0.2 H

Capacitance (C): 0.01 F

Determine the voltage transfer function (H(s)) of the circuit for a range of frequencies from 10 Hz to 100 kHz.

II. Write down the equation for the voltage transfer function (H(S)) of the series RLC circuit in terms of R, L, C and the complex frequency variable

III. Write MATLAB code to:

   a. Define the given circuit parameters (R, L, and C).
   b. Generate a range of frequencies from 10 Hz to 100 kHz.
   c. Calculate the complex frequency variable s for each frequency in the range.
   d. Calculate the voltage transfer function (H(s)) for each frequency using the equation from question II.

IV. Magnitude and Phase Plot: Write MATLAB code to:

   a. Calculate the magnitude and phase of H(s) for the entire frequency range.
   b. Plot the Bode plot of the magnitude (in decibels) and phase (in degrees) of H(s) as a function of frequency.

V. Based on the Bode plot obtained in question IV(b), analyze and discuss the behavior of the series RLC circuit as the frequency changes from 10 Hz to 100 kHz. Explain how the circuit responds to different frequencies and highlight any resonance frequencies if observed.

5. Transient Response of Parallel RL, RC and RLC Circuits

Examine the time varying characteristics of parallel RL, RC and RLC circuits for given values of R, L and C using MATLAB software.

5.1 Transient Response of Parallel RL

1. **Define Circuit Parameters:**
   - Resistance (R) in ohms

- Inductance (L) in henrys
- Applied Voltage (V_{in}) in volts

2. **Define Symbolic Variables:**
   - t as the time variable
   - $I_L(t)$ as the current through the inductor as a function of time

3. **Create the Differential Equation:**

   The differential equation for a parallel RL circuit can be written as:

   $$L\frac{dI_L(t)}{dt} + \frac{V_{in}}{R} = 0$$

   Where:

   - L is the inductance.
   - R is the resistance.
   - $I_L(t)$ is the current through the inductor as a function of time.
   - V_{in} is the applied voltage.

Hints

- **Solve the Differential Equation:**

  Use the '**dsolve**' function to solve the differential equation symbolically:

  HINT

```
% Define the differential equation
syms t
IL = sym('IL(t)');
eqn = L * diff (IL, t) + Vin / R == 0;

% Solve the differential equation symbolically
IL_solution = dsolve (eqn, IL);
IL_solution = simplify(IL_solution);
```

- **Create a Time Vector and Evaluate the Solution:**

  Create a time vector and evaluate $I_L(t)$ for the given time vector:

```
% Create a time vector
t_span = 0:0.01:5*tau; % Adjust the time span as needed

% Evaluate IL(t) for the given time vector
IL_numeric = double (subs (IL_solution, t, t_span));
```

- **Plot the Transient Response:**

Plot the transient response of the current $I_L(t)$ as a function of time:

```
% Plot the transient response
plot (t_span, IL_numeric);
```

```
xlabel ('Time (s)');
ylabel ('Current (A)');
title ('Transient Response of Parallel RL Circuit');
grid on;
```

In this code:

- We define the circuit parameters$(\boldsymbol{R}, \boldsymbol{L}, \boldsymbol{and}\ \boldsymbol{V_{in}})$.
- We create a symbolic variable t for time and a symbolic variable $\boldsymbol{I_L(t)}$ for the current through the inductor.
- We define the differential equation and solve it symbolically.
- We create a time vector for the simulation.
- We evaluate the symbolic solution for $\boldsymbol{I_L(t)}$ numerically.
- Finally, we plot the transient response of the current.

Try

1. **Circuit Parameters:**

   Given the following values:
   Resistor (R): 220 ohms
   Inductor (L): 0.1 H
   Applied Voltage (V_in): 12 V
   Calculate the time constant (τ) of the parallel RL circuit.

2. **Transient Analysis:**

   a) Write down the first-order differential equation governing the transient response of the RL circuit in terms of the current $I_L(t)$ through the inductor.

   b) Solve the differential equation symbolically to obtain the expression for $I_L(t)$ as a function of time (t).

3. **MATLAB Simulation:**

   Write MATLAB code to:

   a) Define the circuit parameters $(\boldsymbol{R}, \boldsymbol{L}, \boldsymbol{and}\ \boldsymbol{V_{in}})$**.**

   b) Simulate the transient response of the RL circuit using the **'lsim'** function. Use a time vector from 0 to 5τ with a suitable time step.

   c) Plot the transient response of the current I_L through the inductor as a function of time. Label the axes appropriately.

4. **Time Constant Verification:**

   a) Calculate the time constant (τ) numerically from the transient response obtained in question 3c.

   b) Compare the numerically calculated time constant with the value obtained in question 1. Explain any differences or similarities.

5. **Analysis:**

Based on the transient response plot obtained in question 3c, discuss the behavior of the current (I_L) in the RL circuit as it responds to the voltage step input. Specifically, explain the time constant, the initial current, and the behavior as time progresses.

5.2 Transient Response of Parallel RC

To analyze the transient response of a parallel RC circuit in MATLAB, you can use the circuit's differential equation and solve it numerically. The transient response describes how the circuit behaves as it moves from one steady state to another when an input voltage or current changes suddenly.
Assuming you have a parallel RC circuit with a voltage source connected in parallel to a resistor (R) and a capacitor (C), you can write the differential equation for this circuit as follows:

$$V(t)=I(t)\cdot R+VC(t)$$

Where:

V(t) is the voltage across the circuit as a function of time.

I(t) is the current through the resistor as a function of time.

$V_C(t)$ is the voltage across the capacitor as a function of time.

You can use MATLAB to solve this differential equation and obtain the transient response.

Hints

1. Define the circuit parameters (R and C) and the input voltage waveform (V_in) as functions of time.

```
R = 1; % Resistor value in ohms
C = 0.1; % Capacitor value in farads

% Define the input voltage as a function of time (for example, a step input)
t = 0:0.01:5; % Time vector
V_in = zeros(size(t)); % Initialize the input voltage vector
V_in (t >= 0) = 1; % Set the input voltage to 1V for t >= 0
```

2. Write the differential equation based on the circuit equation and use MATLAB's ode45 function to solve it numerically.

```
% Define the differential equation for the parallel RC circuit
% dVc/dt = (Vin - Vc) / (R * C)
% Where Vc is the voltage across the capacitor as a function of time
dVc_dt = @(t, Vc) (V_in(t) - Vc) / (R * C);

% Initial condition (at t = 0, Vc = 0)
Vc0 = 0;

% Solve the differential equation using ode45
```

```
[t, Vc] = ode45(dVc_dt, t, Vc0);
```

3. Plot the transient response of the voltage across the capacitor.

```
% Plot the transient response
figure;
plot (t, Vc);
xlabel ('Time (s)');
ylabel ('Voltage across Capacitor (V)');
title ('Transient Response of Parallel RC Circuit');
grid on;
```

4. To simulate and analyze the transient response of a parallel RC (Resistor-Capacitor) circuit in MATLAB, you can use the following syntax:

```
% Define circuit parameters
R = 220; % Resistance in ohms
C = 0.01 % Capacitance in farads
Vin = 12;% Applied voltage in volts

% Calculate the time constant (tau)
tau = R * C;

% Define the symbolic variable for time
syms t

% Define the differential equation for the voltage VC(t) across the capacitor
VC = sym('VC(t)');
eqn = C * diff(VC, t) + VC / R == Vin;

% Solve the differential equation symbolically
VC_solution = dsolve(eqn, VC);
VC_solution = simplify(VC_solution);

% Create a time vector
t_span = 0:0.001:5*tau; % Adjust the time span as needed

% Evaluate VC(t) for the given time vector
VC_numeric = double (subs (VC_solution, t, t_span));

% Plot the transient response
plot (t_span, VC_numeric);
xlabel ('Time (s)');
ylabel ('Voltage across Capacitor (V)');
title ('Transient Response of Parallel RC Circuit');
grid on;
```

Try:

1. **Circuit Parameters:**

   Given the following values:
   Resistor (R): 100 ohms
   Capacitance (C): 0.01 F
   Applied Voltage (V_in): 10 V

Calculate the time constant (τ) of the parallel RC circuit.

2. **Transient Analysis:**
    a) Write down the first-order differential equation governing the transient response of the RC circuit in terms of the voltage (Vc) across the capacitor.
    b) Solve the differential equation symbolically to obtain the expression for Vc(t) as a function of time (t).

3. **MATLAB Simulation:**

Write MATLAB code to:
    a) Define the circuit parameters (R, C, and V_in).
    b) Simulate the transient response of the RC circuit using the **'lsim'** function. Use a time vector from 0 to 5τ with a suitable time step.
    c) Plot the transient response of the voltage (Vc) across the capacitor as a function of time. Label the axes appropriately.

4. **Time Constant Verification:**
    a) Calculate the time constant (τ) numerically from the transient response obtained in question 3c.
    b) Compare the numerically calculated time constant with the value obtained in question 1. Explain any differences or similarities.

5.3 Transient Response of Parallel RLC

To calculate the transient response of a parallel RLC circuit in MATLAB, you can use the circuit's differential equation and solve it numerically. Here's an example of how you can do this:
Assuming you have a parallel RLC circuit with the following parameters:
Resistance (R)
Inductance (L)
Capacitance (C)
Voltage source (V)
Initial conditions (initial voltage across the capacitor and initial current through the inductor)
Here's the MATLAB code to calculate and plot the transient response of the parallel RLC circuit:

Hints

```
 % Define circuit parameters
R = 1;       % Resistance (in ohms)
L = 0.5;     % Inductance (in henries)
C = 0.2;     % Capacitance (in farads)
V = 5;       % Voltage source (in volts)

% Define initial conditions
Vc0 = 0;     % Initial voltage across the capacitor (in volts)
Il0 = 0;     % Initial current through the inductor (in amperes)

% Create a time vector
t = 0:0.01:5; % Time from 0 to 5 seconds with a step of 0.01 seconds
```

```
% Define the differential equation for the parallel RLC circuit
% d2Vc/dt^2 + (1/(R*C)) * dVc/dt + (1/(L*C)) * Vc = (1/L) * dIl/dt
% d2Il/dt^2 + (1/(R*L)) * dIl/dt + (1/(L*C)) * Il = (1/L) * dVc/dt
% Where Vc is the voltage across the capacitor and Il is the current through the
inductor

% Define a function that returns the derivatives of Vc and Il
dVcdt = @(t, Vc, Il) (1/L) * (V - Vc) - (1/(R*C)) * Vc;
dIldt = @(t, Vc, Il) (1/L) * (Vc - V) - (1/(R*L)) * Il;

% Use the ODE solver to solve the system of differential equations
initial_conditions = [Vc0, Il0];
[t, y] = ode45(@(t, y) [dVcdt(t, y(1), y(2)); dIldt(t, y(1), y(2))], t,
initial_conditions);

% Extract voltage across the capacitor and current through the inductor
Vc = y(:, 1);
Il = y(:, 2);

% Plot the transient response
subplot(2, 1, 1);
plot(t, Vc);
xlabel('Time (s)');
ylabel('Voltage across Capacitor (V)');
title('Transient Response of Parallel RLC Circuit');
grid on;

subplot(2, 1, 2);
plot(t, Il);
xlabel('Time (s)');
ylabel('Current through Inductor (A)');
grid on;
```

Try

1. Mathematical Modeling:
   a) Consider a parallel RLC circuit with the following parameters:
      - Resistance (R): 2 ohms
      - Inductance (L): 0.5 henries
      - Capacitance (C): 0.1 farads
      - Voltage source (V): 10 volts

   b) Define the initial conditions as follows:
      - Initial voltage across the capacitor (Vc0): 0 volts
      - Initial current through the inductor (Il0): 0 amperes

   c) Write the differential equations that describe the behavior of this parallel RLC circuit over time. Express these equations in terms of voltage across the capacitor (Vc) and current through the inductor (Il).

2. Using MATLAB, simulate and plot the transient response of the parallel RLC circuit for the following conditions:

   - Time span: 0 to 5 seconds
   - Time step: 0.01 seconds

   You should create MATLAB scripts/functions to do the following:

   a) Define the circuit parameters and initial conditions as specified in Problem 1.
   b) Create a time vector that covers the time span.
   c) Implement the differential equations for the parallel RLC circuit.
   d) Use MATLAB's ODE solver (ode45) to solve the differential equations.
   e) Plot the transient response, showing both the voltage across the capacitor and the current through the inductor on separate graphs.

3. Analysis and Interpretation

   a) Analyze the plots obtained in Problem 2. Explain how the voltage across the capacitor and the current through the inductor change over time during the transient response.
   b) Calculate and report the time constant (τ) of this parallel RLC circuit based on its parameters (R, L, and C). Discuss the significance of the time constant in relation to the transient response.

6. Generation of Three Phase Wave Form

In this code:

- Parameters such as frequency and amplitude are defined.
- Phase differences and phase sequences are specified in degrees.
- A time vector (t) is created to represent time.
- Phase angles in radians are calculated for each phase based on time and phase differences.
- Sinusoidal waveforms for each phase (Phase A, Phase B, and Phase C) are generated.
- Finally, the waveforms are plotted on separate subplots to visualize the three-phase AC waveform.
- You can adjust the 'frequency', 'amplitude', 'phase_diff_AB', 'phase_diff_BC', and 'phase_diff_CA' values to observe different phase differences and phase sequences in the generated waveform.

Hints

```
% Define parameters
frequency = 50;  % Frequency in Hertz
amplitude = 220;  % Amplitude of each phase in volts

% Define phase differences and phase sequences
phase_diff_AB = 0;   % Phase difference between phases A and B (in degrees)
phase_diff_BC = 120; % Phase difference between phases B and C (in degrees)
phase_diff_CA = 240; % Phase difference between phases C and A (in degrees)

% Create time vector
t = 0:0.0001:0.1;  % Time vector (0 to 0.1 seconds with a step of 0.0001 seconds)
```

```
% Calculate phase angles in radians
omega = 2 * pi * frequency;
theta_A = omega * t;
theta_B = omega * t - deg2rad(phase_diff_AB);
theta_C = omega * t - deg2rad(phase_diff_CA);

% Generate sinusoidal waveforms for each phase
phase_A = amplitude * sin(theta_A);
phase_B = amplitude * sin(theta_B);
phase_C = amplitude * sin(theta_C);

% Plot the three-phase waveforms
figure;
subplot (3, 1, 1);
plot (t, phase_A, 'b');
xlabel ('Time (s)');
ylabel ('Phase A Voltage (V)');
title ('Three-Phase AC Waveform');
grid on;

subplot (3, 1, 2);
plot (t, phase_B, 'r');
xlabel ('Time (s)');
ylabel ('Phase B Voltage (V)');
grid on;

subplot (3, 1, 3);
plot (t, phase_C, 'g');
xlabel ('Time (s)');
ylabel ('Phase C Voltage (V)');
grid on;

% Adjust plot settings for better visualization
linkaxes;

% Legend for phase sequence
legend ('Phase A', 'Phase B', 'Phase C');
```

Try

Problem 1: Mathematical Background

Explain the concept of three-phase AC power systems, including the significance of phase differences and phase sequences. Discuss how the phase differences and sequences impact the operation of electrical systems.

Problem 2: MATLAB Programming

Write MATLAB code to generate three-phase AC waveforms with the following specifications:

- Frequency: 50 Hz
- Amplitude: 220 V

You should generate three phases (Phase A, Phase B, and Phase C) with different phase differences and sequences. Use a time vector with a duration of at least 0.1 seconds and a suitable time step.

a) Implement code to create sinusoidal waveforms for each phase with the specified frequency and amplitude.
b) Generate waveforms for the following scenarios:
   - Scenario 1: Balanced three-phase AC with a phase sequence of ABC and zero phase differences.
   - Scenario 2: Balanced three-phase AC with a phase sequence of BCA and phase differences of 120 degrees between phases.
   - Scenario 3: Unbalanced three-phase AC with a phase sequence of CBA and phase differences of 45 degrees between phases.
c) Plot each scenario's waveforms on separate graphs, clearly labeling the phases and including appropriate axis labels and titles.

Problem 3: Analysis and Discussion

a) Analyze and compare the waveforms generated in Scenario 1, Scenario 2, and Scenario 3. Explain how the phase differences and phase sequences affect the shape and behavior of the three-phase AC waveforms.
b) Discuss the practical applications of different phase sequences in electrical systems. Provide examples of where each phase sequence might be used and why.

Submission Instructions:

1. Provide well-commented MATLAB code for generating the three-phase AC waveforms (Problem 2).
2. Write a clear and concise analysis and discussion of the waveforms and phase sequences (Problem 3) in a separate document.
3. Submit both your MATLAB code and the analysis document as your assignment.

7. Three phase measurements

In this code:

- We generate three-phase sinusoidal waveforms with the specified frequency and amplitude. Phase B and Phase C are 120 degrees out of phase with respect to Phase A, representing a balanced three-phase system.
- RMS voltage for each phase is calculated using the '**rms'** function.
- Phase angles between Phase A and other phases (Phase B and Phase C) are calculated using the **'atan2'** function and converted from radians to degrees.
- Total power is calculated for each phase assuming a known impedance, and then the total power is summed to get the total power in the three-phase system.
- You can adjust the **'frequency'**, **'amplitude'**, and **'impedance'** values to match your specific waveform and system parameters. This code will help you determine essential electrical quantities for a three-phase waveform in MATLAB.

Hints

```
% Define the three-phase waveform data
time = 0:0.0001:0.1;  % Time vector (0 to 0.1 seconds with a step of 0.0001 seconds)
```

```
frequency = 50;       % Frequency in Hertz
amplitude = 220;      % Amplitude of each phase in volts

% Generate the three-phase sinusoidal waveforms
omega = 2 * pi * frequency;
phase_A = amplitude * sin(omega * time);
phase_B = amplitude * sin(omega * time - 2*pi/3);  % Phase B is 120 degrees ahead of
Phase A
phase_C = amplitude * sin(omega * time + 2*pi/3);  % Phase C is 120 degrees behind
Phase A

% Calculate RMS voltage for each phase
rms_A = rms(phase_A);
rms_B = rms(phase_B);
rms_C = rms(phase_C);

% Calculate phase angles between Phase A and other phases
phase_angle_B = rad2deg(atan2(imag(phase_B), real(phase_B)));
phase_angle_C = rad2deg(atan2(imag(phase_C), real(phase_C)));

% Calculate total power
power_A = rms_A^2 / impedance;  % Assuming a known impedance
power_B = rms_B^2 / impedance;
power_C = rms_C^2 / impedance;
total_power = power_A + power_B + power_C;

% Display the results
fprintf('RMS Voltage (Phase A): %.2f V\n', rms_A);
fprintf('RMS Voltage (Phase B): %.2f V\n', rms_B);
fprintf('RMS Voltage (Phase C): %.2f V\n', rms_C);
fprintf('Phase Angle (Phase B - Phase A): %.2f degrees\n', phase_angle_B);
fprintf('Phase Angle (Phase C - Phase A): %.2f degrees\n', phase_angle_C);
fprintf('Total Power: %.2f Watts\n', total_power);
```

Try:

Problem 1: Mathematical Background

I. Provide a brief explanation of three-phase electrical systems, including the significance of balanced and unbalanced systems. Discuss key electrical quantities such as RMS voltage, phase angles, and power in the context of three-phase waveforms.

II. Modify the barrel shifter by changing the number of data lines to 16 bit wide to support 16 different functionalities

Problem 2: MATLAB Programming

Write MATLAB code to generate and analyze a three-phase waveform with the following specifications:

- Frequency: 60 Hz
- Amplitude: 220 V (for each phase)
- Duration: 0.2 seconds

a. Generate three sinusoidal waveforms (Phase A, Phase B, and Phase C) with the specified frequency and amplitude. Phase B and Phase C should have phase differences of 120 degrees with respect to Phase A.

b. Calculate and display the following electrical quantities:

- RMS voltage for each phase.
- Phase angles between Phase A and Phase B, and between Phase A and Phase C.
- Total power in the three-phase system. Assume a balanced system and a known impedance.

Problem 3: Analysis and Interpretation

a. Analyze and discuss the results obtained in Problem 2. Explain the significance of RMS voltage, phase angles, and total power in a three-phase system.

b. Consider a scenario where one of the phases (e.g., Phase B) becomes unbalanced and has a different amplitude (e.g., 200 V) compared to the other phases. Modify the MATLAB code to simulate this unbalanced scenario and recalculate the electrical quantities. Discuss how the unbalance affects the system.

Submission Instructions:

1. Provide well-commented MATLAB code for generating and analyzing the three-phase waveform (Problem 2).
2. Write a clear and concise analysis and interpretation of the electrical quantities and the impact of unbalance (Problem 3) in a separate document.
3. Submit both your MATLAB code and the analysis document as your assignment.

8. Virtual instruments (vi) using Labview

Virtual instruments (VIs) in LabVIEW (Laboratory Virtual Instrument Engineering Workbench) are a powerful way to create customized software applications for data acquisition, analysis, and instrument control. VIs are graphical representations of an instrument or a measurement system and can be created using LabVIEW's intuitive programming environment. Here are the steps to create a basic virtual instrument (VI) using LabVIEW:

Hints

```
% Launch LabVIEW
```

Start LabVIEW on your computer.

```
% Open a New VI
```

Click on "File" in the menu bar.

Select "New VI" to create a new VI or "Open VI" if you have an existing VI that you want to work on.

```
% Front Panel Design
```

The front panel is where you design the user interface of your VI. You can add controls (inputs) and indicators (outputs) by using the tools in the "Controls Palette" on the left-hand side of the LabVIEW window.

Drag and drop controls (e.g., buttons, knobs, numeric inputs) onto the front panel to specify user inputs.

Drag and drop indicators (e.g., graphs, numeric displays) onto the front panel to display the results or data.

Arrange and customize the front panel elements to create a user-friendly interface for your VI.

```
% Block Diagram Design
```

The block diagram is where you define the functionality of your VI by connecting wires between the controls and indicators and adding programming logic.

Right-click on the block diagram and use the "Functions Palette" on the left to add functions and structures.

Use wires to connect the controls to functions and indicators to display the results.
Write LabVIEW code to perform the desired operations, calculations, or data processing.

```
% Wiring and Programming
```

Wire controls and indicators together to establish data flow between the front panel and the block diagram. To wire, click and drag from a control to an indicator or to a function's input.

Write LabVIEW code on the block diagram using graphical programming. You can use structures like loops and case structures for more complex operations.

```
% Save the VI
```

Click on "File" and select "Save" to save your VI with a meaningful name and location on your computer.

```
% Run the VI
```

Click the "Run" button on the toolbar to execute your VI. Interact with the controls on the front panel and observe the results on the indicators.

```
% Debug and Test
```

LabVIEW provides debugging tools such as probes, highlight execution, and breakpoints to help you identify and resolve issues in your VI.

```
% Documentation and Deployment
```

Consider documenting your VI with comments and descriptions to make it more understandable for others. You can also create a standalone application or share the VI with others by creating an executable file or a LabVIEW run-time engine package.

Try

Problem 1: Understanding the Existing VI

1. Download the provided LabVIEW VI file (e.g., "Assignment_VI.vi") or use an existing VI if available.
2. Open the VI in LabVIEW.
3. Examine the front panel and block diagram of the VI to understand its functionality and user interface.

Problem 2: Editing and Building the Existing VI

1. Edit the existing VI to achieve the following:
   - Modify the user interface by adding at least one additional control (e.g., a numeric input, a switch, a slider).
   - Modify the VI's functionality by adding LabVIEW code to perform a specific operation (e.g., mathematical calculations, data processing).
2. Build and run the edited VI to ensure it works as intended.

Problem 3: Creating a Sub VI

1. Identify a portion of the LabVIEW code in the edited VI that can be modularized into a separate sub VI. This portion should be a self-contained task or operation.
2. Create a new VI (Sub VI) that performs the identified task or operation. This sub VI should have inputs and outputs as needed.
3. Incorporate the Sub VI into the edited VI by replacing the relevant LabVIEW code with a call to the Sub VI.
4. Build and run the edited VI with the Sub VI to verify that the modularization works correctly.

Problem 4: Documentation and Presentation

1. Document your changes and explain the functionality of the edited VI and the purpose of the Sub VI in a clear and concise report.
2. Create a brief presentation (e.g., slides or a video) demonstrating the edited VI, the Sub VI, and the modifications you made. Explain the benefits of modularization using Sub VIs.

Submission Instructions:

1. Submit the edited LabVIEW VI file (with your modifications).
2. Submit the created Sub VI as a separate LabVIEW file.
3. Submit a report (in PDF or Word format) that includes documentation of your changes and explanations.
4. Provide a link or file for your presentation materials.

9. Generation of Common Wave Forms Using Labview

Generating common waveforms using LabVIEW is relatively straightforward with the built-in functions and structures provided by the LabVIEW programming environment. You can create waveforms such as sine waves, square waves, triangular waves, and saw tooth waves using LabVIEW. Below are the steps and syntax to generate some of these common waveforms:

Hints

Generating a Sine Wave:

1. Place a while loop on the block diagram to continuously generate the sine wave.
2. Inside the loop, use a Numeric Control to adjust the frequency of the sine wave.
3. Use a Sine Waveform function to generate the sine wave.

```
/**   Generating a Sine Wave:*/

Sine Waveform VI -> Amplitude (constant) = 1.0 -> Frequency (control) = [frequency control value]
```

Generating a Square Wave:

1. Place a while loop on the block diagram.
2. Inside the loop, use a Numeric Control to adjust the frequency of the square wave.
3. Use a Square Waveform function to generate the square wave.

```
/**   Generating a Square Wave: */

Square Waveform VI -> Amplitude (constant) = 1.0 -> Frequency (control) = [frequency control value]
```

Generating a Triangular Wave:

1. Place a while loop on the block diagram.
2. Inside the loop, use a Numeric Control to adjust the frequency of the triangular wave.
3. Use a Triangular Waveform function to generate the triangular wave.

```
/**   Generating a Triangulate Wave: */

Triangular Waveform VI -> Amplitude (constant) = 1.0 -> Frequency (control) = [frequency control value]
```

Generating a Saw tooth Wave:

1. Place a while loop on the block diagram.
2. Inside the loop, use a Numeric Control to adjust the frequency of the saw tooth wave.
3. Use a Saw tooth Waveform function to generate the saw tooth wave.

```
        /**   Generating a Saw Tooth Wave: */

Saw tooth Waveform VI -> Amplitude (constant) = 1.0 -> Frequency (control) = [frequency control value]
```

Try

1 Display of maximum and minimum values of a sinusoidal signal.

2 Display of modulation of two sinusoidal signals.

10. Frequency measurement using Lissajous figures in Lab View

Frequency measurement using Lissajous figures in LabVIEW involves the use of an oscilloscope or waveform display to analyze the Lissajous pattern formed when two sinusoidal signals with different frequencies are plotted against each other. The frequency measurement can be performed by counting the number of intersections or lobes of the Lissajous figure within a known time period. Below is a step-by-step guide on how to implement frequency measurement using Lissajous figures in LabVIEW:

Hints

```
/** Create a LabVIEW VI*/

Open LabVIEW and create a new VI (Virtual Instrument).

/** Generate Lissajous Figures*/
Generate two sinusoidal signals with different frequencies and amplitudes using
appropriate Signal Generation functions. Use separate Frequency controls for each
signal.
Create two separate Waveform Charts or Graphs on the front panel.
Wire the generated signals to the Waveform Charts or Graphs.

/** Set Up the Timebase */
Add a Numeric Control for the timebase (the time interval over which you want to
measure frequency).
Connect the Numeric Control to a Wait (ms) function to introduce a delay in your loop.

/** Count Lissajous Lobe Intersections */

Place a While Loop on the block diagram.
Inside the loop, increment a counter (initialized to zero) every time you detect an
intersection or lobe crossing of the Lissajous figure.
Compare the elapsed time (using the Tick Count function) with the desired timebase, and
exit the loop when the desired time has passed.

/** Calculate Frequency */

Outside the loop, calculate the frequency using the following formula:
Frequency (Hz) = Number of Intersections / Timebase (seconds)

/** Display the Frequency */

Create a Numeric Indicator to display the calculated frequency.
Connect the calculated frequency to the Numeric Indicator.

/** Run the VI */

Run the VI and observe the Lissajous figure on the waveform charts.
The calculated frequency will be displayed on the Numeric Indicator when the VI
completes.
```

Try:

1. Display of Lissajious pattern for sinusoidal voltages of same frequencies.
2. Display of Lissajious pattern for sinusoidal voltages of same different frequencies.

11. Structures Using Labview

Analyze the virtual instrumentation (VI) using control loops, arrays, charts and graphs in LabVIEW software

Hints

```
/** Open LabVIEW and create a new VI */

/** Place the following controls on the front panel */
   - Numeric Control for Frequency (Hz)
   - Numeric Control for Amplitude (V)
   - Numeric Control for Duration (seconds)
   - Waveform Chart (for displaying the original signal)
   - Waveform Graph (for displaying the FFT result)

/** Create a FOR loop on the block diagram with the following elements inside it */
   - Numeric Constant (0) wired to the FOR loop iteration terminal
   - Numeric Control for Duration wired to the count terminal of the FOR loop

/** Inside the FOR loop, generate a sinusoidal signal using the Sine Waveform function
*/
   - Place a Sine Waveform function inside the FOR loop
   - Wire the Frequency control to the "Frequency (Hz)" input of the Sine Waveform
function
   - Wire the Amplitude control to the "Amplitude (V)" input of the Sine Waveform
function
   - Use the FOR loop iteration terminal to generate a time signal (0, 1 sample period,
2 sample periods, etc.)
   - Wire the time signal to the "Time (s)" input of the Sine Waveform function

/** Wire the output of the Sine Waveform function to a Build Array function to create
an array of the generated signal */

/** After the FOR loop, add a Peak Detector VI to detect peaks in the signal */
   - Place a Peak Detector VI on the block diagram
   - Wire the output array to the "Input Array" input of the Peak Detector
   - Set the "Threshold" parameter of the Peak Detector VI to a suitable value

/** Connect the output of the Peak Detector VI to the "Data" input of the Waveform
Chart to display the original signal with peaks highlighted */

/** Place a Waveform Chart Clear function outside the FOR loop and wire it to the chart
to clear the chart before displaying the new data */

/** Create a WHILE loop with the following elements inside it */
   - A Waveform Graph Clear function to clear the graph for displaying FFT results
   - An FFT function to compute the Fast Fourier Transform of the signal:
     - Wire the output array (from the Peak Detector) to the "Time Domain Signal" input
of the FFT function
   - A waveform chart to display the FFT result:
     - Wire the output of the FFT function to the "Amplitude Spectrum" input of the
waveform chart

/** Wire the output of the Waveform Chart Clear function inside the WHILE loop to clear
the chart before displaying the new data */

/** Run the VI to generate the sinusoidal signal, detect peaks, compute the FFT, and
display the results on the charts */

/** Analyze the displayed results, including peak positions and frequency components in
```

```
the FFT result */

This LabVIEW example demonstrates the use of FOR and WHILE loops, charts (waveform chart and waveform graph), arrays, and analysis VIs (Peak Detector and FFT) in a signal processing application. Students can modify and extend this example to explore more complex signal processing and analysis tasks.
```

Try:

1. Obtain VI using For loop and While loop.
2. Obtain VI using charts and arrays.
3. Obtain VI using graphs.

12. Simulation of low pass and high pass filters using digital simulation

To plot the characteristics of low-pass and high-pass filters using MATLAB, you can follow these steps and use the **'freqz'** function for plotting the frequency response:

Hints

```
/**   Design the Filters */

Design low-pass and high-pass filters using the 'fir1' or 'fdesign' functions. Here's an example using 'fir1':

% Design a low-pass filter
lowpass_order = 50; % Filter order
lowpass_cutoff_frequency = 0.2; % Cutoff frequency (normalized)
lowpass_filter = fir1(lowpass_order, lowpass_cutoff_frequency);

% Design a high-pass filter
highpass_order = 50; % Filter order
highpass_cutoff_frequency = 0.2; % Cutoff frequency (normalized)
highpass_filter = fir1(highpass_order, highpass_cutoff_frequency, 'high');

/** Plot the Frequency Response */

Plot the frequency response of the filters using the 'freqz' function:

% Frequency response of the low-pass filter
figure;
freqz (lowpass_filter, 1, 1024);
title ('Low-Pass Filter Frequency Response');
xlabel ('Normalized Frequency (\pi radians/sample)');
ylabel ('Magnitude (dB)');

% Frequency response of the high-pass filter
figure;
freqz (highpass_filter, 1, 1024);
title ('High-Pass Filter Frequency Response');
xlabel ('Normalized Frequency (\pi radians/sample)');
ylabel ('Magnitude (dB)');
```

```
In this code:
'freqz (lowpass_filter, 1, 1024)' calculates and plots the frequency response of the low-pass filter.
'freqz (highpass_filter, 1, 1024)' calculates and plots the frequency response of the high-pass filter.

/** Display the Phase Response (Optional) */
If you also want to display the phase response, you can modify the 'freqz' function as follows:

% Phase response of the low-pass filter
figure;
[h, w] = freqz (lowpass_filter, 1, 1024);
plot (w, unwrap(angle(h)));
title ('Low-Pass Filter Phase Response');
xlabel ('Normalized Frequency (\pi radians/sample)');
ylabel ('Phase (radians)');

% Phase response of the high-pass filter
figure;
[h, w] = freqz (highpass_filter, 1, 1024);
plot (w, unwrap(angle(h)));
title ('High-Pass Filter Phase Response');
xlabel ('Normalized Frequency (\pi radians/sample)');
ylabel ('Phase (radians)');
```

Try

Problem 1: Filter Design

1. Design a low-pass filter with the following specifications:
    - Filter Order: 40
    - Cutoff Frequency: 0.2 (normalized frequency)
2. Design a high-pass filter with the following specifications:
    - Filter Order: 30
    - Cutoff Frequency: 0.3 (normalized frequency)

Problem 2: Frequency Response Plot

1. Plot the magnitude frequency response (in dB) of both the low-pass and high-pass filters.
    - Use the **'freqz'** function to calculate and plot the frequency response.
    - Label the x-axis as "Normalized Frequency (\pi radians/sample)" and the y-axis as "Magnitude (dB)."
2. Plot the phase frequency response (in radians) of both filters.
    - Use the freqz function to calculate and plot the phase response.
    - Label the x-axis as "Normalized Frequency (\pi radians/sample)" and the y-axis as "Phase (radians)."

Problem 3: Filtered Signal

1. Generate a noisy input signal using MATLAB. You can use the **'randn'** function to generate random noise and the **'sin'** function to generate a sinusoidal signal.
2. Apply both the low-pass and high-pass filters to the noisy signal to obtain filtered signals.
3. Plot the noisy input signal, the low-pass filtered signal, and the high-pass filtered signal on the same graph.

Problem 4: Frequency Response Analysis

1. Analyze the frequency response plots and explain how the magnitude and phase characteristics of the filters align with their designed specifications
2. Describe the differences in filtering results between the low-pass and high-pass filters. Discuss which frequencies are attenuated and which are passed through.

Problem 5: Submission

1. Prepare a report summarizing your filter design, frequency response analysis, and filtered signal results.
2. Include the MATLAB code used for filter design, frequency response plotting, and signal processing in your report.
3. Provide explanations and interpretations of the results in the report.

13. Sensor Circuit Using LAB View

Designing the electric and electronic circuit of a sensor in LabVIEW involves creating a virtual representation of the circuit using LabVIEW's graphical programming interface. You'll use various components, such as controls, indicators, and wiring, to simulate the behavior of the physical sensor circuit. Below is a simplified example of how you can design a virtual sensor circuit in LabVIEW.

HINTS

```
/** Open LabVIEW and Create a New VI */

        Launch LabVIEW and create a new VI (Virtual Instrument).

/** Design the Front Panel */

        Add controls and indicators to the front panel to represent components in your
sensor circuit. For example:

Numeric Controls for resistors, capacitors, or other components with adjustable values.
Switches or buttons to simulate the opening or closing of switches.
Numeric Indicators or Waveform Charts to display sensor readings.
You can find these controls and indicators in the "Controls" and "Indicators" palettes.

/** Create the Block Diagram */

Switch to the block diagram by clicking on the "Block Diagram" tab.

/** Wire Components Together */

Use wiring to connect the controls and indicators to simulate connections in your
sensor circuit.
For example, use wires to connect a voltage source (control) to a resistor (indicator).

/** Implement Logic */

Use LabVIEW's graphical programming to implement the logic and calculations needed for
your sensor circuit. You can use functions and structures like loops, case structures,
and math functions.

/** Simulate Sensor Behavior */

Implement code that simulates the behavior of the sensor. For instance, if you're
simulating a temperature sensor, write code to generate temperature readings based on
user inputs or predefined conditions.

/** Display Sensor Output */

Use indicators (such as numeric indicators or waveform charts) to display the simulated
sensor readings on the front panel.

/** Run the VI */

Save and run your VI to see the simulated behavior of the sensor circuit.

Adjust control values on the front panel to see how they affect the sensor output.
```

```
/** Analyze and Validate */

Analyze the behavior of your virtual sensor circuit to ensure it behaves as expected.

Validate the circuit's response to different inputs and conditions.

/** Document Your Design*/

Document your virtual sensor circuit design, including circuit diagrams, code explanations, and any assumptions you've made.

/** Extend and Customize */

Depending on your needs, you can extend the virtual sensor circuit to include more components, sensors, or complex behavior.
```

Try

Problem 1: Front Panel Design

1. Open LabVIEW and create a new VI.
2. On the front panel, design the user interface to represent an electric and electronic circuit of a sensor. Include the following elements:
   - Numeric Controls for adjusting resistor values (e.g., R1, R2).
   - Toggle Switches to simulate the opening and closing of switches.
   - Numeric Indicators to display voltage, current, or other relevant values.
   - A Waveform Chart to display dynamic sensor data.
   - Any additional elements you consider relevant to your sensor circuit.

Problem 2: Block Diagram Implementation

1. Switch to the block diagram and use LabVIEW's graphical programming to simulate the behavior of the sensor circuit. Implement the following:
   - Wiring to connect the components as per the designed circuit.
   - Logic for calculating voltages, currents, or sensor readings based on component values and switch positions.
   - Simulation of time-dependent behavior if applicable (e.g., transient response).

Problem 3: Simulate Sensor Operation

1. Create a scenario where the sensor circuit interacts with an external environment or input signal. For example, if simulating a temperature sensor, simulate temperature changes over time.
2. Implement code that calculates and updates sensor data based on the interaction with the environment or input signal.

Problem 4: Real-Time Monitoring

1. Configure LabVIEW to provide real-time monitoring of sensor data.
    - Use a loop structure to continuously update sensor readings.
    - Display the sensor data in real-time on the Waveform Chart on the front panel.

Problem 5: Data Analysis and Validation

1. Run the VI and observe the behavior of the virtual sensor circuit.
    - Verify that the circuit responds correctly to changes in component values and switch positions.
    - Analyze the data displayed on the Waveform Chart and ensure it matches the expected behavior.

Problem 6: Report and Documentation

1. Prepare a report that includes the following:
    - A description of the designed sensor circuit.
    - A discussion of how LabVIEW was used to simulate its behavior.
    - Results and observations from running the VI.
    - An analysis of the data collected during the simulation.
2. Include screenshots of the LabVIEW front panel and block diagram in your report.

Problem 7: Presentation

Prepare a brief presentation to demonstrate your virtual sensor circuit and discuss your findings.

14. Sensor Circuit Using LAB View

Measuring the speed of a machine using a proximity sensor in LabVIEW typically involves using a sensor to detect the position of a rotating object and then calculating the speed based on the change in position over time.

HINTS

```
/** Hardware Setup */

Connect your proximity sensor to the appropriate data acquisition hardware (e.g., a DAQ card) and ensure that it is properly configured.

/** Open LabVIEW and Create a New VI */

Launch LabVIEW and create a new VI (Virtual Instrument).

/** Front Panel Design */

On the front panel, add the necessary controls and indicators:
        Numeric Indicator for displaying the speed.
        A Start Button to initiate speed measurement.
        A Graph or Chart for visualizing speed data (optional).
```

```
        Any other controls or indicators you may need.

/** Create the Block Diagram */

Switch to the block diagram by clicking on the "Block Diagram" tab.

/** Initialize Variables */

Use a Numeric control to set the initial position of the machine.

/** While Loop for Measurement */

Place a While Loop on the block diagram to continuously measure and update the speed.

Use a Case Structure with the Stop Button to control the loop's execution.

/** Read Proximity Sensor Data */

Inside the While Loop, use DAQ functions or appropriate libraries to read data from the
proximity sensor.

Store the sensor's position data in a variable.

/** Calculate Speed */

Calculate the speed of the machine by determining the change in position over time
(velocity).

You can use LabVIEW's Math functions to perform this calculation.

/** Display Speed */

Display the calculated speed on the Numeric Indicator on the front panel.

/** Optional: Data Visualization */

If desired, you can also plot the speed data on a Graph or Chart in real-time to
visualize the machine's speed profile.

/** Start and Stop Controls */

Use the Start Button to initiate speed measurement by enabling the While Loop.

Use the Stop Button to stop the measurement by disabling the While Loop.

/** Run the VI */

Save and run your VI to measure and display the speed of the machine.
```

This is a simplified example, and in practice, you may need to consider additional factors such as calibration, noise filtering, and error handling. The specific implementation details can vary depending on the type of proximity sensor and data acquisition hardware you are using.

Try

Problem 1: Hardware Setup

Set up a proximity sensor and connect it to a DAQ (Data Acquisition) device or hardware.

Problem 2: LabVIEW VI Design

1. Open LabVIEW and create a new VI.
2. Design the front panel with the following elements:
    - A numeric indicator to display the measured speed (in RPM or other appropriate units).
    - A start button to initiate speed measurement.
    - A stop button to stop the measurement.
    - A chart or graph indicator to display the speed profile (optional but recommended).
    - Any additional controls or indicators you consider necessary.

Problem 3: Block Diagram Implementation

1. Create a While Loop on the block diagram.
2. Use a Case Structure inside the loop to handle the start and stop buttons. The loop should only execute when the start button is pressed and stop when the stop button is pressed.
3. Use DAQ functions or libraries (appropriate for your hardware) to read data from the proximity sensor. You may need to configure the DAQ task or device settings.
4. Calculate the speed of the machine based on the sensor data. Consider the appropriate formula for converting sensor data into speed (e.g., RPM calculation).
5. Display the calculated speed on the numeric indicator on the front panel.
6. If desired, plot the speed data on the chart or graph indicator for real-time visualization.

Problem 4: User Interface

1. Test the VI by running it.
    - Click the start button to initiate speed measurement.
    - Observe the speed reading on the numeric indicator.
    - Use the stop button to halt the measurement.

Problem 5: Data Analysis

1. Analyze the data collected during the measurement.
   - Calculate the average speed.
   - Identify any variations or anomalies in the speed profile.

Problem 6: Report

1. Prepare a report summarizing your LabVIEW program's design and functionality.
    - Include screenshots of the LabVIEW front panel and block diagram.
    - Describe the hardware setup and connection.

Problem 7: Presentation

1. Create a brief presentation to demonstrate your LabVIEW program and discuss your findings.
    - Discuss how the program measures machine speed.

TEXT BOOKS:

1 A Chakrabarthy, “Circuit Theory”, Dhanpat Rai Publications, 6^{th} edition, 2006.

2 A Sudhakar, Shyammohan S Palli, “Circuits and Networks”, Tata McGraw Hill, 4^{th} edition, 2010.

References

1. William Hayt, Jack E Kemmerly S.M. Durbin, “Engineering Circuit Analysis”, Tata McGraw Hill, 7^{th} edition, 2010.
2. K S Suresh Kumar, “Electric Circuit Analysis”, Pearson Education, 1^{st} edition, 2013.
3. Rudrapratap, “Getting started with MATLAB: A Quick Introduction for Scientists and Engineers”, OxfordUniversity Press, 1^{st} edition, 19994.

DC MACHINES LABORATORY

III Semester: EEE								
Course Code	Category	Hours / Week			Credits	Maximum Marks		
AEEC09	Core	L	T	P	C	CIA	SEE	Total
		-	-	3	1.5	40	60	100
Contact Classes: Nil	Tutorial Classes: Nil	Practical Classes: 45				Total Classes: 45		
Prerequisite: There are no prerequisites to take this course								

I. COURSE OVERVIEW:

This laboratory course is to meet the requirements of practical work meant for basic operation, analysis and design of electrical machines. It provides hands-on experience by examining the electrical and mechanical characteristics of various DC machines. Analyze the characteristics of DC machines and separate the various losses in electrical machines by conducting different tests.

II. COURSES OBJECTIVES:

The students will try to learn

I. The elementary experimental and modelling skills for handling problems with electrical machines in the industries and domestic applications to excel in professional career.

II. The operation of DC Machines and its role in power transmission and distribution.

III. The intuitive knowledge needed to test and analyze the performance leading to design of electric machines by conducting various tests and calculate the performance parameters.

III. COURSE OUTCOMES:

At the end of the course students should be able to:

CO 1: Analyze the performance characteristics of dc machine under various loading conditions.
CO 2: Determine the critical field resistance and speed of dc shunt generator using open circuit characteristics.
CO 3: Examine the performance of DC shunt machine with different speed control techniques and predetermine the efficiency.
CO 4: Estimate and separate the core losses in dc machine by conducting a suitable test.
CO 5: Examine the performance and speed control of dc machines using simulation tools.

DO's

1. Once the operation is completed pull the plug itself rather chord attached to it.
2. To repair the equipment switch-off the supply and go on.
3. To operate the equipment on supply, see that hands are dry, if that is not possibly hide the hand in the pockets.
4. If a person comes in contact with current unexpectedly don't touch the person with hands but immediately use any insulator material and shut down the power (like leather belts, wood and plastic bars etc).
5. If water is nozzles on the equipment, immediately shunt down the power using circuit breaker or pull out the plug.
6. Use the connecting wires of good continuity, short circuit of connecting wire leads damage of circuit parameters

DON'Ts

1. Do not wear loose clothing and do not hold any conducting materials in contact with skin when the power is on.
2. Do not pull out the connections until unless all the currents are dead.

3. Do not wait for switches to de-magnetize when there is a delay but pull out the plug.
4. Do not overload the circuit by plugging in too many appliances.
5. If you are mentally and physically stressed don’t operate the power equipment.
6. Never operate the equipment under wet conditions.
7. Do not interconnect two or more wires, take appropriate length of wire.

SAFETY NORMS

1. The lab must be equipped with fire extinguisher.
2. See that the connections are made tight.
3. Use single plug for each equipment.
4. Cover the body completely to avoid arc effect.
5. To change the connections during the experiment, switch off the supply and carry on.
6. Used equipment may get heated, so take care handling the equipment after it is used.
7. Do the wiring, all set ups and check the circuit connections before the supply is on

IV. COURSE SYLLABUS:

EXERCISES FOR ELECTRICAL DC MACHINES LABORATORY

Note: Students are encouraged to bring their own laptops for laboratory practice session

1. OPEN CIRCUIT CHARACTERISTICS OF DC SHUNT GENERATOR.

Develop the circuit for analyzing the characteristics of DC shunt generator

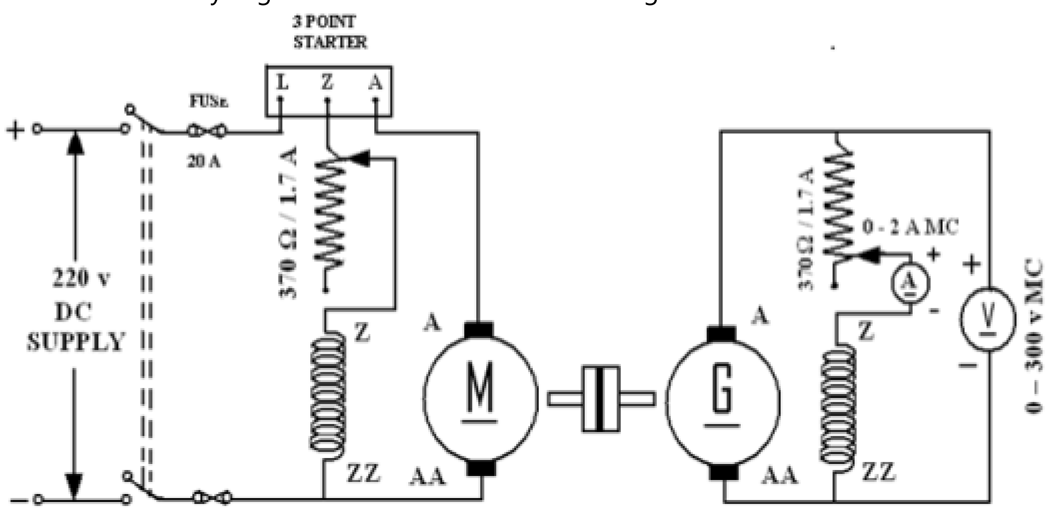


Figure 1 - MAGNETIZATION CHARACTERISTIC

Try

1. From the Open circuit characteristics calculate the critical resistance of field winding.
2. Using magnetization characteristics calculate the critical speed of DC shunt generator at 100 ohms in figure 1.
3. Determine the performance of DC generator using the magnetization curve.
4. Calculate the critical value of shunt field resistance at 1500 rpm.

2. LOAD TEST ON DC SHUNT GENERATOR

Design the DC shunt generator circuit under full, 3/4th, half and 1/4th load conditions for analyzing the performance of the machine

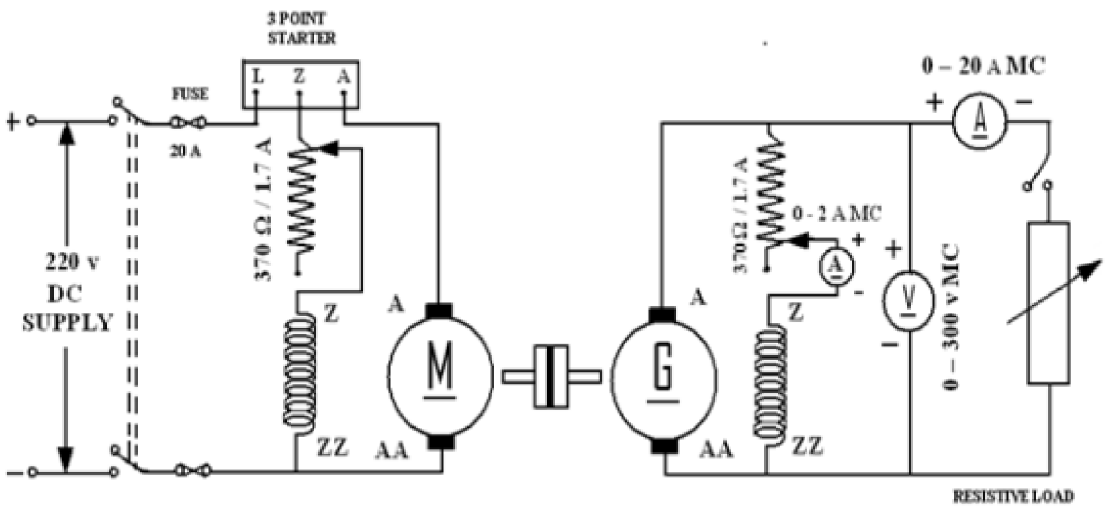


Figure 2 - LOAD TEST ON DC SHUNT GENERATOR

Try

1. Calculate the different armature currents (i.e ia equal il plus if) for shunt generator under various loads in figure 2.
2. Draw the External characteristics using armature currents (0 to 13 A) with respect to the load currents.
3. Draw the Internal characteristics using generated induced emf (0 to 220V) with respect to the field currents (0 to 2A).

3. LOAD TEST ON DC SERIES GENERATOR

Design the DC series generator circuit under full, 3/4th, half and 1/4th load conditions for analyzing the performance of the machine

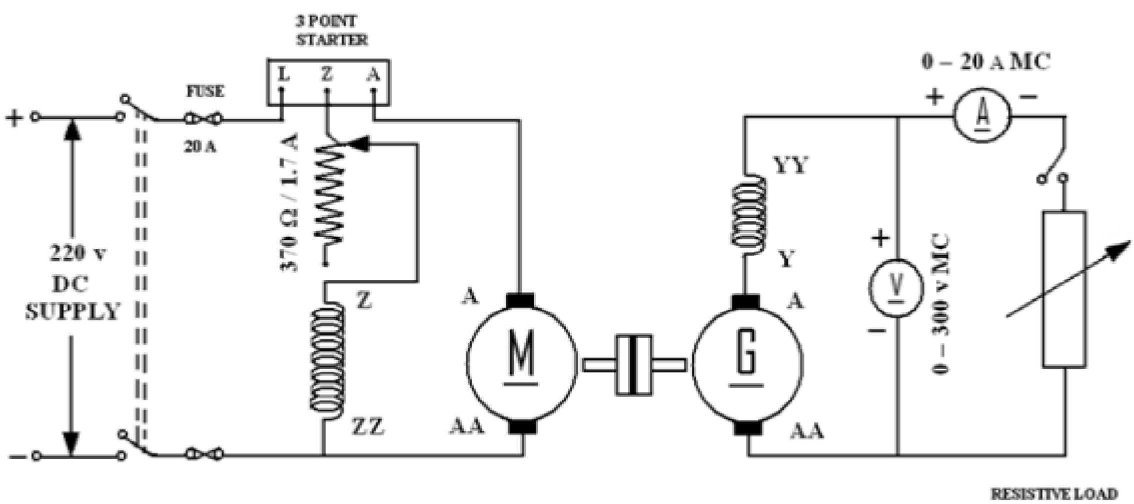


Figure 3 - LOAD TEST ON DC SERIES GENERATOR

Try

1. Calculate the different armature currents (i.e I_a equal I_a plus if) for series generator under various Loads In figure 3

2. Draw the External characteristics using armature currents (0 to 13 A) with respect to the load currents.
3. Draw the Internal characteristics using generated induced emf (0 to 220V) with respect to the field currents (0 to 2A).

4. LOAD TEST ON DC COMPOUND GENERATOR

Design the DC compound generator circuit under full, 3/4th, half and 1/4th load conditions for analyzing the performance of the machine

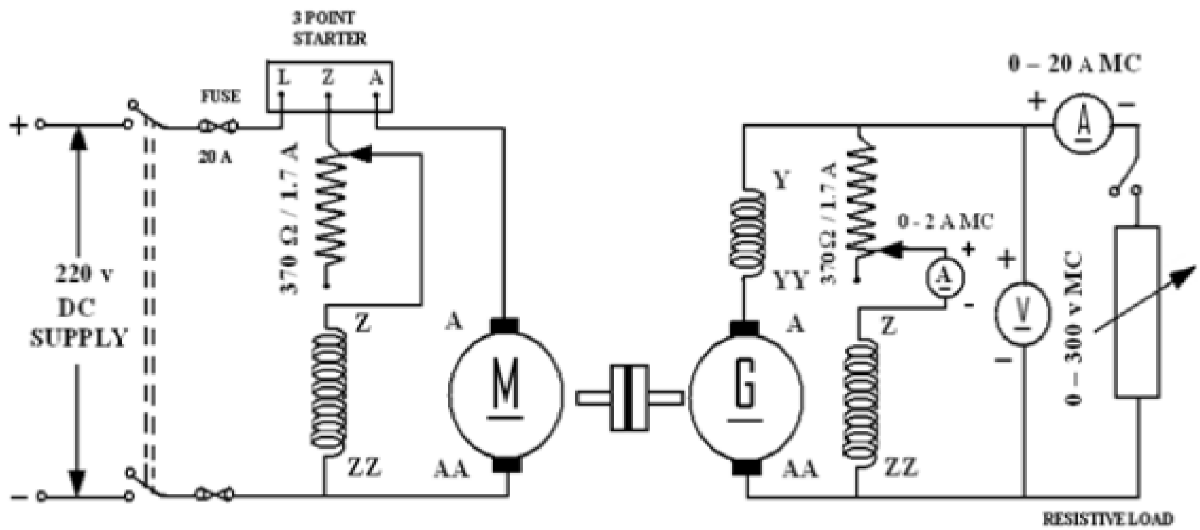


Figure 4 - LOAD TEST ON DC COMPOUND GENERATOR

Try

1. Calculate the different armature currents (i.e ia equal il plus if) for compound generator under various loads in figure 4
2. Draw the External characteristics using armature currents (0 to 13 A) with respect to the load currents.
3. Draw the Internal characteristics using generated induced emf(0 to 220V) with respect to the field currents (0 to 2A).

5. HOPKINSON'S TEST

Develop a method of testing for two identical dc shunt machines which are mechanically coupled and also electrically connected in parallel

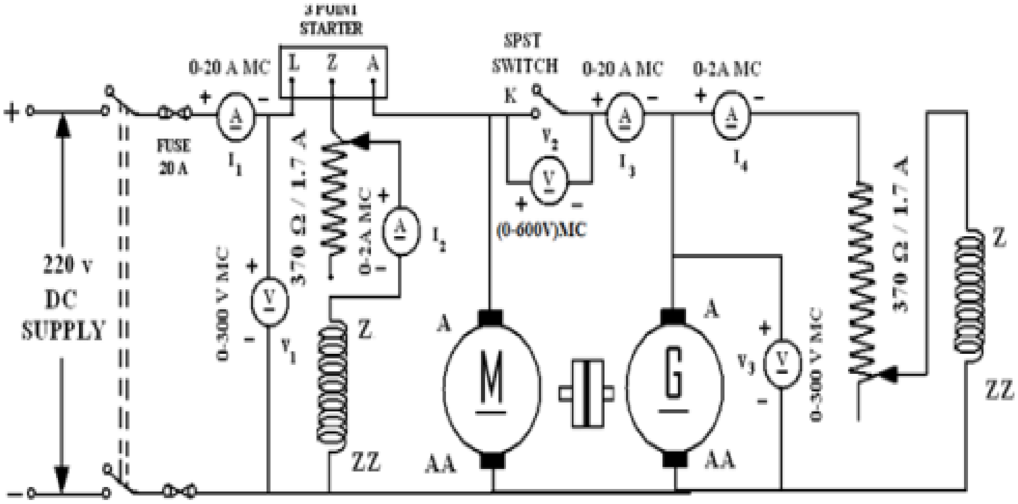


Figure 5 - Identical DC Shunt Machines

Try

1. Calculate the efficiency of two identical dc shunt machines if the armature resistance of each machine is 0.025 ohms, line voltage of 230V and line current excluding both the field currents 2A, motor armature current 10A, field current 1A and 2A figure 5.
2. Draw the performance characteristics of two identical dc shunt machines.
3. Find the iron losses depend on the emf generated in the armature

6. FIELD'S TEST

1. Develop a method of testing for two similar dc series machines depend on the accuracy with which the motor input and generator output are measured
2. Determine load or unknown current through a R_4 resistor using Thevenin's equivalent circuit.

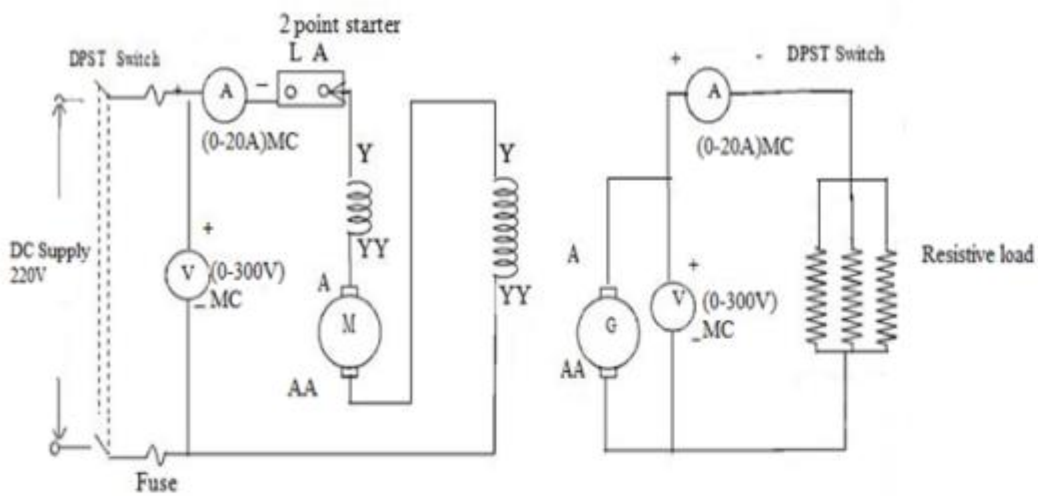


Figure 6 - Identical DC Shunt Machines

Try

1. Calculate the efficiency of both dc series machines if the armature resistance of each machine is 2 ohms in figure 6
2. Draw the performance characteristics of two dc series machines.
3. Find the no load rotational losses of both the machines and total losses in the whole set.

7. SWINBURNE'S TEST AND SPEED CONTROL OF DC SHUNT MOTOR

1. Design the suitable test under no load conditions to measure no load losses inDc shunt machines and speed control of DC shunt motor.

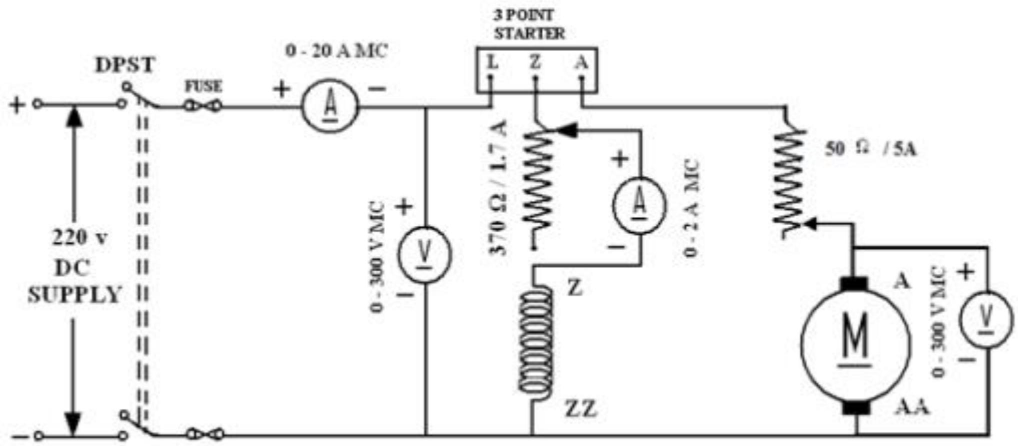


Figure 7 - Speed control of DC motor

Try

1. Calculate the output power and efficiency when motor takes 10A on full load and 5A on half Load in figure 7.
2. Measure the no load machine losses by using indirect method of testing.
3. Perform the speed control by varying the armature circuit resistance and field circuit resistance of DC shunt motor.

8. BRAKE TEST ON DC COMPOUND MOTOR

Develop the circuit for conducting brake test on DC compound motor.

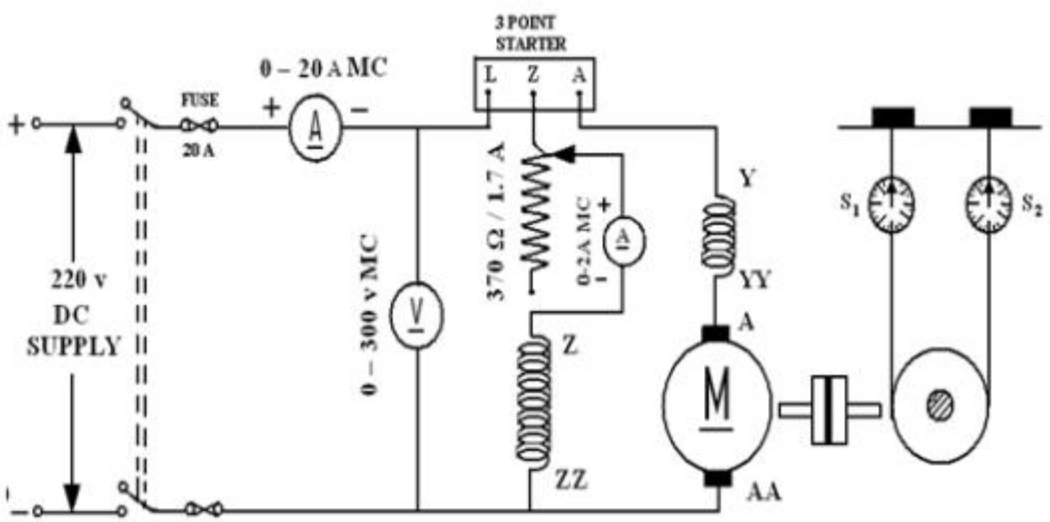


Figure 8 - Brake test on DC compound motor

Try

1. Calculate the efficiency of DC compound motor under different load conditions in figure 8
2. Calculate the shaft torque and shaft power at rated load.
3. Determine the mechanical output power under different weights.

9. BRAKE TEST ON DC SHUNT MOTOR

Develop the circuit for conducting brake test on DC shunt motor

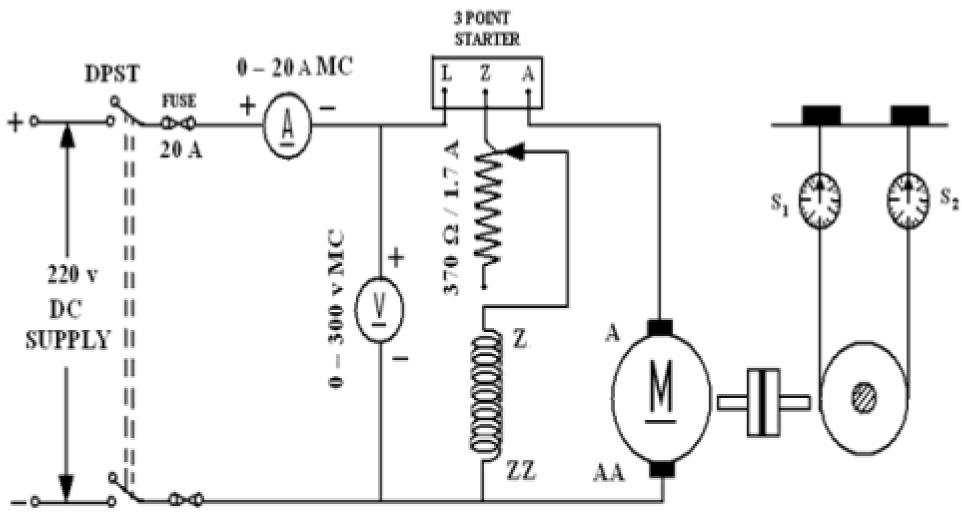


Figure 9 - Brake test on DC shunt motor

Try

1. Calculate the efficiency of DC shunt motor under different load conditions.
2. Calculate the shaft torque and shaft power at rated load.
3. Determine the mechanical output power under different weights

10. RETARDATION TEST

Develop the test for separating the mechanical losses of the DC shunt machine

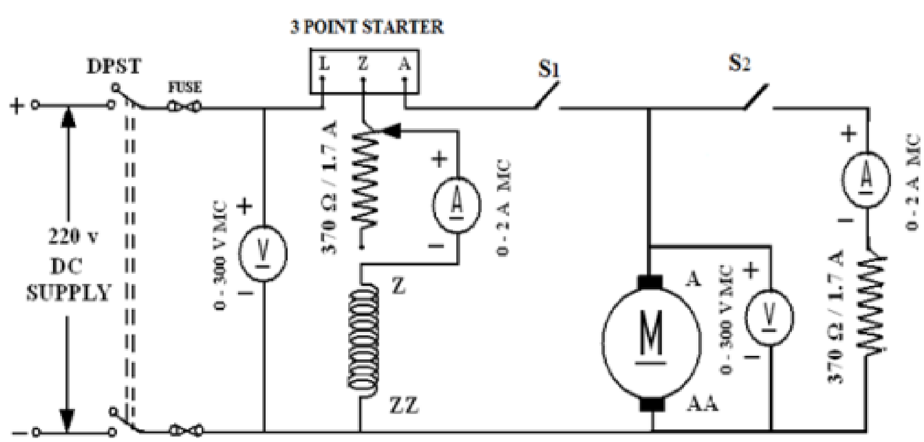


Figure 10 - Retardation Test

Try

1. Find the rotational losses i.e. friction, wind age and iron losses.
2. Measure the moment of inertia of the armature under normal speed of a DC machine 1000rpm.
3. Calculate the efficiency of DC shunt machine when time taken for the speed to fall from 1030 rpm to 970 rpm is 15 seconds with field normally excited in figure 10.

11. SEPARATION OF LOSSES IN DC SHUNT MOTOR

Design the circuit for separating the iron losses in DC shunt motor.

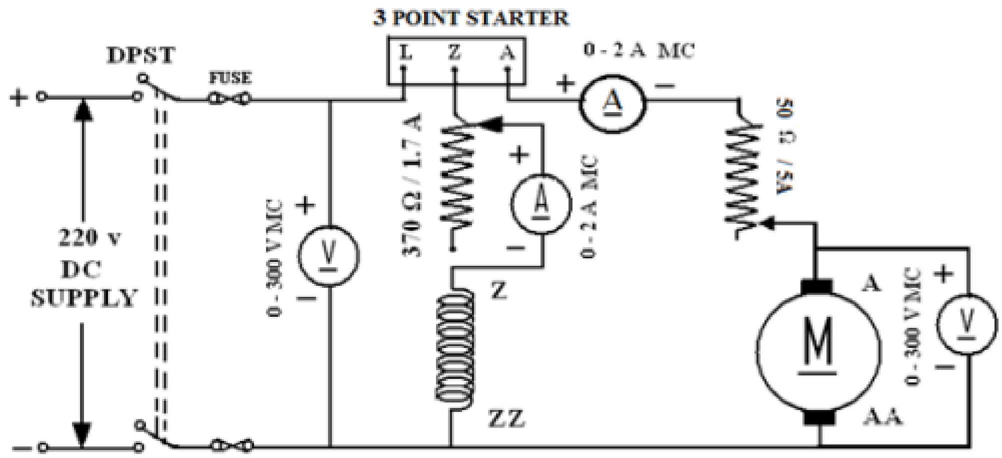


Figure 11 - Figure 9 - separation of core losses

Try

1. Calculate the hysteresis and eddy current losses in Dc shunt motor.
2. Draw the curve for total iron losses for various field currents.

12. MAGNETIZATION CHARACTERISTICS OF DC SHUNT GENERATOR USING DIGITAL SIMULATION

1. Develop the circuit for analyzing the magnetization characteristics of DC shunt generator using Digital simulation.

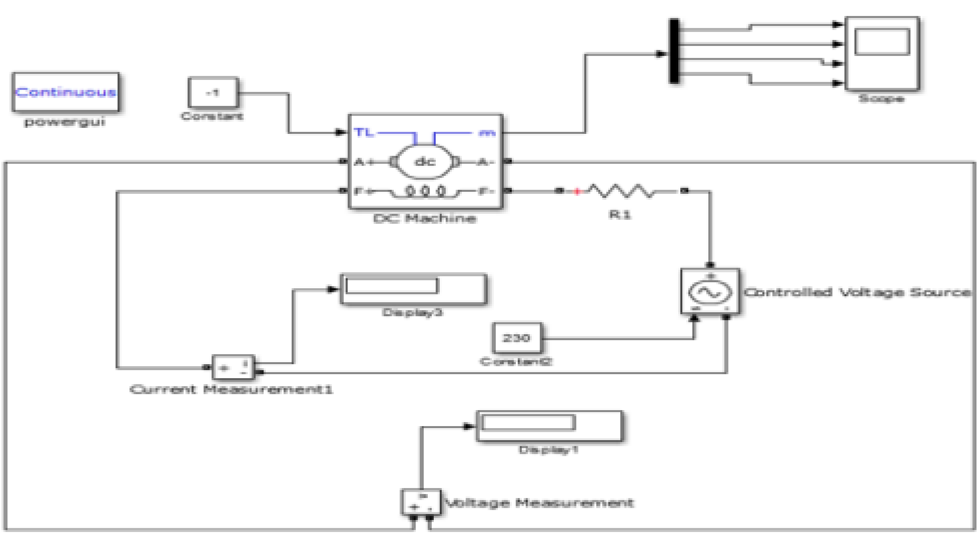


Figure – 12 separation of core losses

Try

1. From the Open circuit characteristics calculate the critical resistance of field winding.
2. Using magnetization characteristics calculate the critical speed of DC shunt generator at 100 ohms.
3. Determine the performance of DC generator using the magnetization curve.
4. Calculate the critical value of shunt field resistance at 1500 rpm

13. LOAD TEST ON DC SHUNT GENERATOR USING DIGITAL SIMULATION

Design the DC shunt generator circuit under full, 3/4th, half and 1/4th load conditions for analyzing the performance of the machine using Digital simulation.

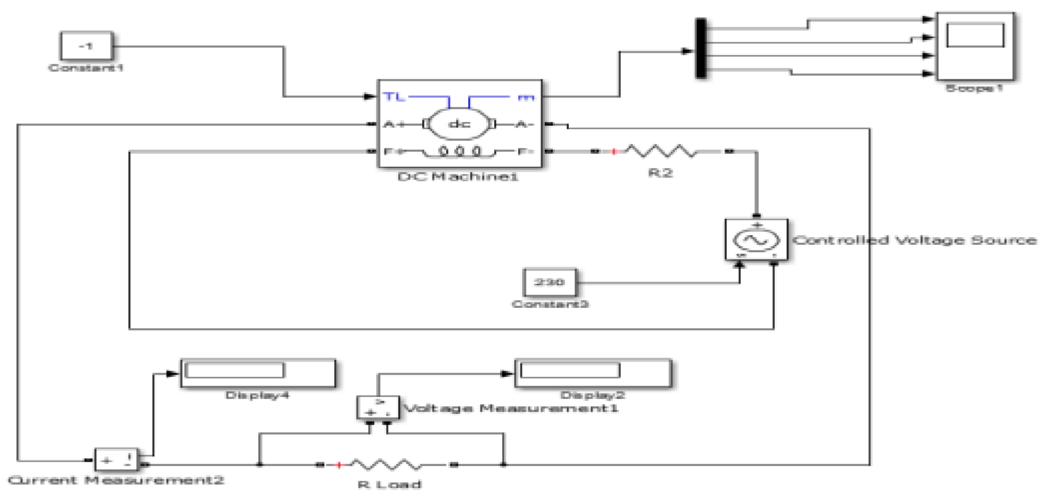


Figure – 13 separation of core losses

Try

1. Calculate the different armature currents (i.e i_a equal i_l plus i_f) for shunt generator under various loads.
2. Draw the External characteristics using armature currents (0 to 13 A) with respect to the load currents
3. Draw the Internal characteristics using generated induced emf (0 to 220V) with respect to the field currents (0 to 2A).

14. SPEED CONTROL OF DC SHUNT MOTOR USING DIGITAL SIMULATION

Design the suitable test for speed control of DC shunt motor using MATAB.

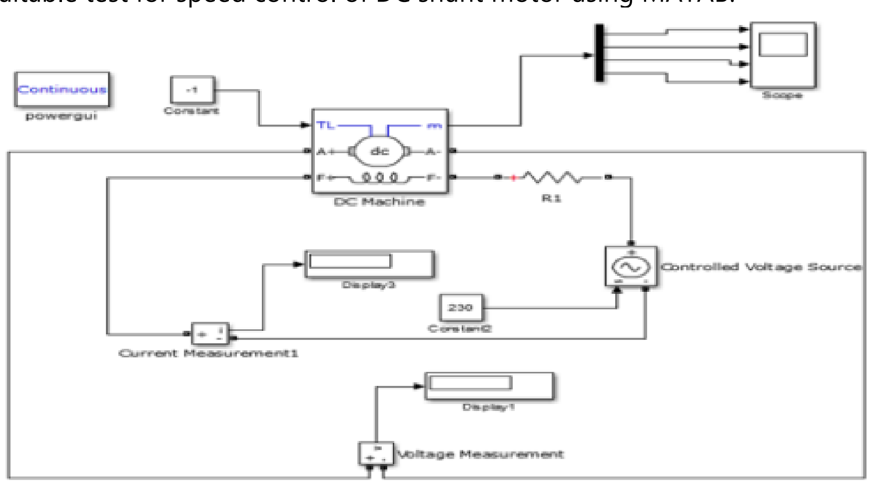


Figure – 14 speed control for various loads

Try

1. Perform the speed control by varying the armature circuit resistance of DC shunt motor.
2. Perform the speed control by varying the field circuit resistance of DC shunt motor.

V. TEXT BOOKS:

1. A Chakrabarti, "*Circuit Theory*", Dhanpat Rai Publications, 8th Edition, 2021.
2. William Hayt, Jack E Kemmerly S.M. Durbin, "*Engineering Circuit Analysis*", Tata McGraw Hill, 9th Edition, 2020.

VI. REFERENCE BOOKS:

1. CL Wadhwa, *Electrical Circuit Analysis including Passive Network Synthesis*, International, 2nd Edition, 2009.
2. David A Bell, *Electric circuits*, Oxford University Press, 7th Edition, 2009.

VII. ELECTRONICS RESOURCES:

1. https://www.allaboutcircuits.com/textbook/
2. https://onlinecourses.nptel.ac.in/noc22_ee93/preview
3. https://www.iare.ac.in

VIII. MATERIALS ONLINE

1. Course template
2. Lab manual

DATA STRUCTURES LABORATORY

<table>
<tr><td colspan="9">III Semester: Common for all branches</td></tr>
<tr><th>Course Code</th><th>Category</th><th colspan="3">Hours / Week</th><th>Credits</th><th colspan="3">Maximum Marks</th></tr>
<tr><td rowspan="2">ACSC10</td><td rowspan="2">Core</td><td>L</td><td>T</td><td>P</td><td>C</td><td>CIA</td><td>SEE</td><td>Total</td></tr>
<tr><td>0</td><td>0</td><td>3</td><td>1.5</td><td>30</td><td>70</td><td>100</td></tr>
<tr><td>Contact Classes: Nil</td><td>Tutorial Classes: Nil</td><td colspan="4">Practical Classes: 45</td><td colspan="3">Total Classes: 45</td></tr>
<tr><td colspan="9">Prerequisite: Programming for Problem Solving using C and Python Programming</td></tr>
</table>

I. COURSE OVERVIEW:

The course covers some of the general-purpose data structures and algorithms, and software development. Topics covered include managing complexity, analysis, static data structures, dynamic data structures and hashing mechanisms. The main objective of the course is to teach the students how to select and design data structures and algorithms that are appropriate for problems that they might encounter in real life. This course reaches to student by power point presentations, lecture notes, and lab which involve the problem solving in mathematical and engineering areas.

II. COURSES OBJECTIVES:

The students will try to learn

I. The skills needed to understand and analyze performance trade-offs of different algorithms / implementations and asymptotic analysis of their running time and memory usage.

II. The basic abstract data types (ADT) and associated algorithms: stacks, queues, lists, tree, graphs, hashing and sorting, selection and searching.

III. The fundamentals of how to store, retrieve, and process data efficiently.

III. COURSE OUTCOMES:

At the end of the course students should be able to:

CO 1	Interpret the complexity of algorithm using the asymptotic notations.
CO 2	Select appropriate searching and sorting technique for a given problem.
CO 3	Construct programs on performing operations on linear and nonlinear data structures for organization of a data
CO 4	Make use of linear data structures and nonlinear data structures solving real time applications.
CO 5	Describe hashing techniques and collision resolution methods for efficiently accessing data with respect to performance.
CO 6	Compare various types of data structures; in terms of implementation, operations and performance.

IV. COURSE CONTENT:

EXERCISES FOR DATA STRUCTURES LABORATORY

Note: Students are encouraged to bring their own laptops for laboratory practice sessions.

1. Getting Started Exercises

1.1 Implicit Recursion

A specific type of recursion called **implicit recursion** occurs when a function calls itself without making an explicit recursive call. This can occur when a function calls another function, which then calls the original code once again and starts a recursive execution of the original function.

Using implicit recursion find the second-largest elements from the array.

In this case, the **find_second_largest** method calls the **find_largest()** function via implicit recursion to locate the second-largest number in a provided list of numbers. Implicit recursion can be used in this way to get the second-largest integer without having to write any more code

Input: nums = [1, 2, 3, 4, 5]

Output: 4

```
def find_largest(numbers):
     # Write code here
     …

def find_second_largest(numbers):
    # Write code here
     …

# Driver code
numbers = [1, 2, 3, 4, 5]

# Function call
second_largest = find_second_largest(numbers)
print(second_largest)
```

1.2 Towers of Hanoi

Tower of Hanoi is a mathematical puzzle where we have three rods (A, B, and C) and N disks. Initially, all the disks are stacked in decreasing value of diameter i.e., the smallest disk is placed on the top and they are on rod A. The objective of the puzzle is to move the entire stack to another rod (here considered C), obeying the following simple rules:

- Only one disk can be moved at a time.
- Each move consists of taking the upper disk from one of the stacks and placing it on top of another stack i.e. a disk can only be moved if it is the uppermost disk on a stack.
- No disk may be placed on top of a smaller disk.

Input: 2

Output: Disk 1 moved from A to B

Disk 2 moved from A to C
Disk 1 moved from B to C

Input: 3

Output: Disk 1 moved from A to C

Disk 2 moved from A to B
Disk 1 moved from C to B

Disk 3 moved from A to C

Disk 1 moved from B to A

Disk 2 moved from B to C

Disk 1 moved from A to C

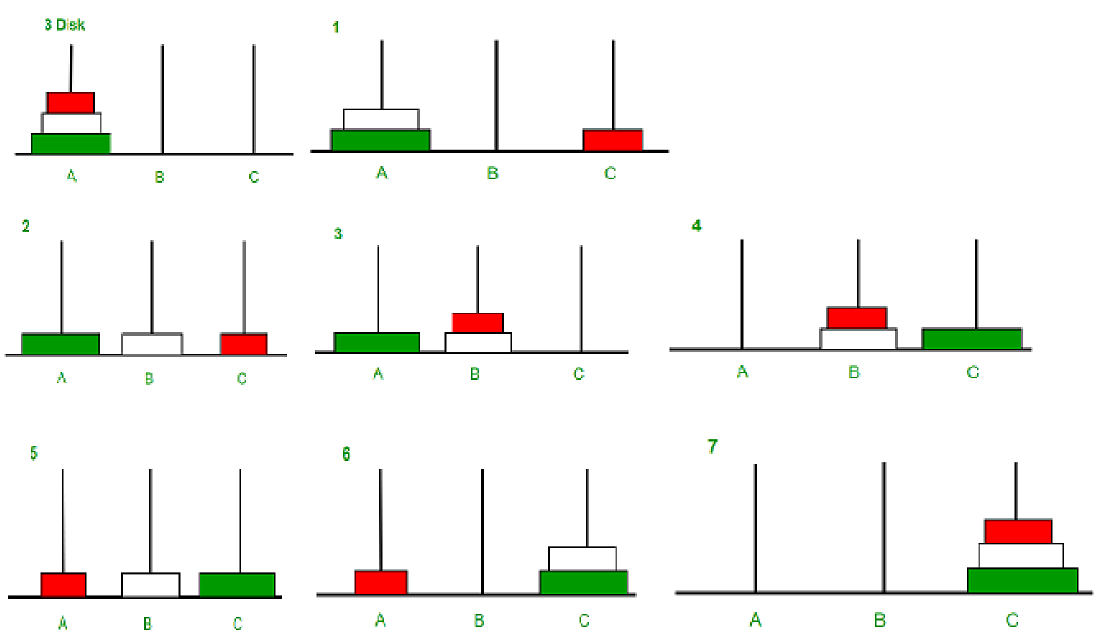
Tower of Hanoi using Recursion:

The idea is to use the helper node to reach the destination using recursion. Below is the pattern for this problem:

- Shift 'N-1' disks from 'A' to 'B', using C.
- Shift last disk from 'A' to 'C'.
- Shift 'N-1' disks from 'B' to 'C', using A.

Follow the steps below to solve the problem:

- Create a function towerOfHanoi where pass the N (current number of disk), from_rod, to_rod, aux_rod.
- Make a function call for N – 1 th disk.
- Then print the current the disk along with from_rod and to_rod
- Again make a function call for N – 1 th disk.



```python
# Recursive Python function to solve Tower of Hanoi
def TowerOfHanoi(n, from_rod, to_rod, aux_rod):
    if n == 0:
        return
    # Write code here
    …

# Driver code
N = 3

# A, C, B are the name of rods
TowerOfHanoi(N, 'A', 'C', 'B')
```

1.3 Recursively Remove all Adjacent Duplicates

Given a string, recursively remove adjacent duplicate characters from the string. The output string should not have any adjacent duplicates.

Input: s = "azxxzy"

Output: "ay"

Explanation:

- First "azxxzy" is reduced to "azzy".
- The string "azzy" contains duplicates
- So it is further reduced to "ay"

Input: "caaabbbaacdddd"

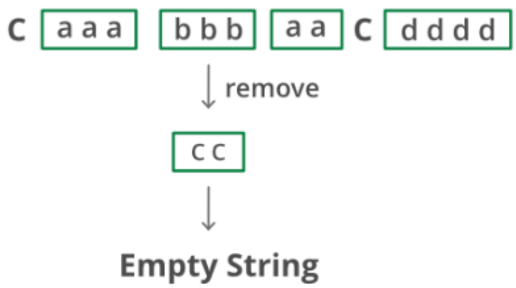
Output: Empty String

Input: "acaaabbbacdddd"

Output: "acac"

Procedure to remove duplicates:

- Start from the leftmost character and remove duplicates at left corner if there are any.
- The first character must be different from its adjacent now. Recur for string of length n-1 (string without first character).
- Let the string obtained after reducing right substring of length n-1 be rem_str. There are three possible cases
  - If first character of rem_str matches with the first character of original string, remove the first character from rem_str.
  - If remaining string becomes empty and last removed character is same as first character of original string. Return empty string.
  - Else, append the first character of the original string at the beginning of rem_str.
- Return rem_str.




```
# Program to remove all adjacent duplicates from a string

# Recursively removes adjacent duplicates from str and returns
# new string. last_removed is a pointer to last_removed character

def removeUtil(string, last_removed):

      # Write code here
       …
def remove(string):
    # Write code here
   …
# Utility functions
def toList(string):
    x = []
    for i in string:
        x.append(i)
    return x

def toString(x):
    return ''.join(x)

# Driver program
string1 = "azxxxzy"
print remove(string1)

string2 = "caaabbbaac"
print remove(string2)

string3 = "gghhg"
print remove(string3)

string4 = "aaaacddddcappp"
print remove(string4)

string5 = "aaaaaaaaaa"
print remove(string5)
```

1.4 Product of Two Numbers using Recursion

Given two numbers x and y find the product using recursion.

Input: x = 5, y = 2

Output: 10

Input: x = 100, y = 5

Output: 500

Procedure

1. If x is less than y, swap the two variables value
2. Recursively find y times the sum of x
3. If any of them become zero, return 0

```
# Find Product of two Numbers using Recursion

# recursive function to calculate multiplication of two numbers
def product( x , y ):
    # Write code here
    …
# Driver code
x = 5
y = 2
print( product(x, y))
```

1.5 Binary to Gray Code using Recursion

Given the Binary code of a number as a decimal number, we need to convert this into its equivalent Gray Code. Assume that the binary number is in the range of integers. For the larger value, we can take a binary number as string.

In gray code, only one bit is changed in 2 consecutive numbers.

Input: 1001

Output: 1101

Explanation: 1001 -> 1101 -> 1101 -> 1101

Input: 11

Output: 10

Explanation: 11 -> 10

Procedure:

The idea is to check whether the last bit and second last bit are same or not, if it is same then move ahead otherwise add 1.

Follow the steps to solve the given problem:

binary_to_grey(n)
if n == 0

    grey = 0;
else if last two bits are opposite to each other
    grey = 1 + 10 * binary_to_gray(n/10))
else if last two bits are same
    grey = 10 * binary_to_gray(n/10))

```
# Convert Binary to Gray code using recursion
# Function to change Binary to Gray using recursion
def binary_to_gray(n):
    # write code here
    …
# Driver Code
binary_number = 1011101
print(binary_to_gray(binary_number), end='')
```

1.6 Count Set-bits of a number using Recursion

Given a number N. The task is to find the number of set bits in its binary representation using recursion.

Input: 21

Output: 3

Explanation: 21 represented as 10101 in binary representation

Input: 16

Output: 1

Explanation: 16 represented as 10000 in binary representation

Procedure:

1. First, check the LSB of the number.
2. If the LSB is 1, then we add 1 to our answer and divide the number by 2.
3. If the LSB is 0, we add 0 to our answer and divide the number by 2.
4. Then we recursively follow step 1 until the number is greater than 0.

```
# Find number of set bits in a number
# Recursive function to find number of set bits in a number
def CountSetBits(n):
    # write code here
    …
# Driver code
n = 21;
# Function call
print(CountSetBits(n));
```

1.7 Fibonacci Series in Reverse Order using Recursion

Given an integer N, the task is to print the first N terms of the Fibonacci series in reverse order using Recursion.

Input: N = 5

Output: 3 2 1 1 0

Explanation: First five terms are – 0 1 1 2 3

Input: N = 10

Output: 34 21 13 8 5 3 2 1 1 0

The idea is to use recursion in a way that keeps calling the same function again till N is greater than 0 and keeps on adding the terms and after that starts printing the terms.

Follow the steps below to solve the problem:

1. Define a function fibo (int N, int a, int b) where
   i. N is the number of terms and
   ii. a and b are the initial terms with values 0 and 1.
2. If N is greater than 0, then call the function again with values N-1, b, a+b.
3. After the function call, print a as the answer.

```
# Function to print the Fibonacci series in reverse order.
def fibo(n, a, b):
    # write code here
    …
# Driver Code
N = 10
fibo(N, 0, 1)
```

1.8 Length of Longest Palindromic Sub-string using Recursion

Given a string S, the task is to find the length longest sub-string which is a palindrome.

Input: S = "aaaabbaa"

Output: 6

Explanation: Sub-string "aabbaa" is the longest palindromic sub-string.

Input: S = "banana"

Output: 5

Explanation: Sub-string "anana" is the longest palindromic sub-string.

The idea is to use recursion to break the problem into smaller sub-problems. In order to break the problem into two smaller sub-problems, compare the start and end characters of the string and recursively call the function for the middle substring.

```
# Find the length of longest palindromic sub-string using Recursion
```

```python
# Function to find maximum of the two variables
def maxi(x, y):
    if x > y:
        return x
    else:
        return y
 # Function to find the longest palindromic substring: Recursion
def longestPalindromic(strn, i, j, count):
    # write code here
    …
# Function to find the longest palindromic sub-string
def longest_palindromic_substr(strn):
    # write code here
    …
strn = "aaaabbaa"
# Function Call
print(longest_palindromic_substr(strn))
```

1.9 Find the Value of a Number Raised to its Reverse

Given a number N and its reverse R. The task is to find the number obtained when the number is raised to the power of its own reverse

Input : N = 2, R = 2

Output: 4

Explanation: Number 2 raised to the power of its reverse 2 gives 4 which gives 4 as a result after performing modulo 10^9+7

Input: N = 57, R = 75

Output: 262042770

Explanation: 57^{75} modulo 10^9+7 gives us the result as 262042770

```python
# Function to return ans with modulo
def PowerOfNum(N, R):
    # write code here
     …
# Driver code
N = 57
R = 75
# Function call
print(int(PowerOfNum(N, R)))
```

1.10 Mean of Array using Recursion

Find the mean of the elements of the array.

Mean = (Sum of elements of the Array) / (Total no of elements in Array)

Input: 1 2 3 4 5

Output: 3

Input: 1 2 3

Output: 2

To find the mean using recursion assume that the problem is already solved for N-1 i.e. you have to find for n

Sum of first N-1 elements = (Mean of N-1 elements) * (N-1)

Mean of N elements = (Sum of first N-1 elements + N-th elements) / (N)

```
# Program to find mean of array
# Function definition of findMean function
def findMean(A, N):
    # write code here
    …
# Driver Code
Mean = 0
A = [1, 2, 3, 4, 5]
N = len(A)
print(findMean(A, N))
```

Try:

1. Given two numbers **N** and **r**, find the value of $^{N}C_{r}$ using recursion.

$$C(n,r) = C(n-1,r-1) + C(n-1,r)$$

Input: N = 5, r = 2

Output: 10

Explanation: The value of 5C2 is 10

2. Predict the output of the following program. What does the following fun() do in general?

```
fp = 15
def fun(n):
    global fp
    if (n <= 2):
        fp = 1
        return 1

    t = fun(n - 1)
    f = t + fp
    fp = t
    return f
```

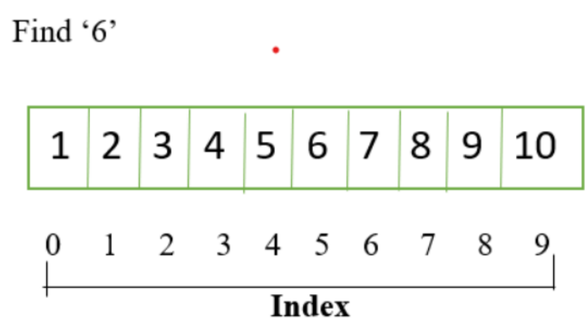
```
# Driver code
print(fun(5))
```

3. **Tail recursion:** Calculate factorial of a number using a Tail-Recursive function.

2. Searching

2.1 Linear / Sequential Search

Linear search is defined as the searching algorithm where the list or data set is traversed from one end to find the desired value. Given an array arr[] of n elements, write a recursive function to search a given element x in arr[].



Note : We find '6' at index '5' through linear search

Linear search procedure:

1. Start from the leftmost element of arr[] and one by one compare x with each element of arr[]

2. If x matches with an element, return the index.

3. If x doesn't match with any of the elements, return -1.

Input: arr[] = {10, 20, 80, 30, 60, 50, 110, 100, 130, 170}
x = 110;
Output: 6
Element x is present at index 6

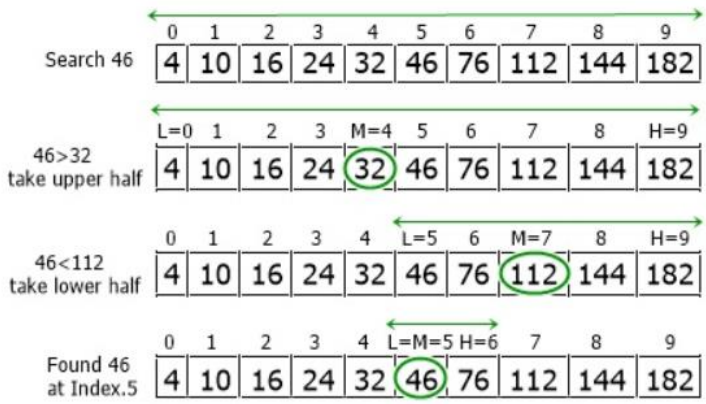
Input: arr[] = {10, 20, 80, 30, 60, 50, 110, 100, 130, 170}
x = 175;
Output: -1
Element x is not present in arr[].

```
# Recursive linear search
def linear_search(arr, curr_index, key):
    # write code here
    …

# Driver code
arr = [10, 20, 80, 30, 60, 50, 110, 100, 130, 170]
x = 110
```

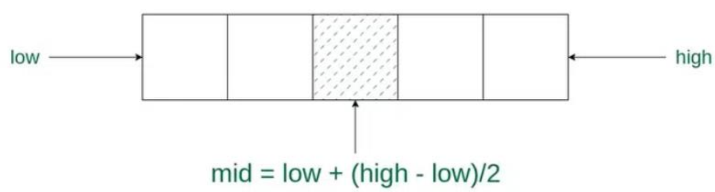
```
linear_search(arr, 0, x)
```

2.2 Binary Search

Binary Search is defined as a searching algorithm used in a sorted array by repeatedly dividing the search interval in half. The idea of binary search is to use the information that the array is sorted and reduce the time complexity to O(log N).



Conditions for Binary Search algorithm:
1. The data structure must be sorted.
2. Access to any element of the data structure takes constant time.



Binary Search Procedure:
1. Divide the search space into two halves by finding the middle index "mid".
2. Compare the middle element of the search space with the key.
3. If the key is found at middle element, the process is terminated.
4. If the key is not found at middle element, choose which half will be used as the next search space.
    a. If the key is smaller than the middle element, then the left side is used for next search.
    b. If the key is larger than the middle element, then the right side is used for next search.
5. This process is continued until the key is found or the total search space is exhausted.

Input: arr = [2, 5, 8, 12, 16, 23, 38, 56, 72, 91]
Output: target = 23
Element 23 is present at index 5

```
# Program for recursive binary search.

# Returns index of x in arr if present, else -1
def binarySearch(arr, l, r, x):
    # write code here
```

```
    …

# Driver Code
arr = [2, 3, 4, 10, 40]
x = 10
result = binarySearch(arr, 0, len(arr)-1, x)

if result != -1:
    print("Element is present at index", result)
else:
    print("Element is not present in array")
```

2.3 Uniform Binary Search

Uniform Binary Search is an optimization of Binary Search algorithm when many searches are made on same array or many arrays of same size. In normal binary search, we do arithmetic operations to find the mid points. Here we precompute mid points and fills them in lookup table. The array look-up generally works faster than arithmetic done (addition and shift) to find the mid-point.

Input: array = {1, 3, 5, 6, 7, 8, 9}, v=3

Output: Position of 3 in array = 2

Input: array = {1, 3, 5, 6, 7, 8, 9}, v=7

Output: Position of 7 in array = 5

The algorithm is very similar to Binary Search algorithm, the only difference is a lookup table is created for an array and the lookup table is used to modify the index of the pointer in the array which makes the search faster. Instead of maintaining lower and upper bound the algorithm maintains an index and the index is modified using the lookup table.

```
# Implementation of above approach

MAX_SIZE = 1000

# lookup table
lookup_table = [0] * MAX_SIZE

# create the lookup table for an array of length n
def create_table(n):
    # write code here
    …
# binary search

def binary(arr, v):
    # write code here
    …

# Driver code
arr = [1, 3, 5, 6, 7, 8, 9]
n = len(arr)
```

```
# create the lookup table
create_table(n)

# print the position of the array
print("Position of 3 in array = ", binary(arr, 3))
```

2.4 Interpolation Search

Interpolation search works better than Binary Search for a Sorted and Uniformly Distributed array. Binary search goes to the middle element to check irrespective of search-key. On the other hand, Interpolation search may go to different locations according to search-key. If the value of the search-key is close to the last element, Interpolation Search is likely to start search toward the end side. Interpolation search is more efficient than binary search when the elements in the list are uniformly distributed, while binary search is more efficient when the elements in the list are not uniformly distributed.

Interpolation search can take longer to implement than binary search, as it requires the use of additional calculations to estimate the position of the target element.

Input: arr = [1, 2, 3, 4, 5, 6, 7, 8, 9]
Output: target = 5

```
# Interpolation search
def interpolation_search(arr, target):
    # write code here
    …


# Driver code
arr = [1, 2, 3, 4, 5, 6, 7, 8, 9]
target = 5

index = interpolation_search(arr, target)
if index == -1:
    print(f"{target} not found in the list")
else:
    print(f"{target} found at index {index}")
```

2.5 Fibonacci Search

Given a sorted array arr[] of size n and an element x to be searched in it. Return index of x if it is present in array else return -1.

Input: arr[] = {2, 3, 4, 10, 40}, x = 10
Output: 3
Element x is present at index 3.

Input: arr[] = {2, 3, 4, 10, 40}, x = 11
Output: -1
Element x is not present.

Fibonacci Search is a comparison-based technique that uses Fibonacci numbers to search an element in a sorted array.

Fibonacci Numbers are recursively defined as F(n) = F(n-1) + F(n-2), F(0) = 0, F(1) = 1. First few Fibonacci Numbers are 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, …

Fibonacci Search Procedure:

Let the searched element be x. The idea is to first find the smallest Fibonacci number that is greater than or equal to the length of the given array. Let the found Fibonacci number be fib (m'th Fibonacci number). We use (m-2)'th Fibonacci number as the index (If it is a valid index). Let (m-2)'th Fibonacci Number be i, we compare arr[i] with x, if x is same, we return i. Else if x is greater, we recur for subarray after i, else we recur for subarray before i.

Let arr[0..n-1] be the input array and the element to be searched be x.

1. Find the smallest Fibonacci number greater than or equal to n. Let this number be fibM [m'th Fibonacci number]. Let the two Fibonacci numbers preceding it be fibMm1 [(m-1)'th Fibonacci Number] and fibMm2 [(m-2)'th Fibonacci Number].

2. While the array has elements to be inspected:

   i. Compare x with the last element of the range covered by fibMm2

   ii. If x matches, return index

   iii. Else If x is less than the element, move the three Fibonacci variables two Fibonacci down, indicating elimination of approximately rear two-third of the remaining array.

   iv. Else x is greater than the element, move the three Fibonacci variables one Fibonacci down. Reset offset to index. Together these indicate the elimination of approximately front one-third of the remaining array.

3. Since there might be a single element remaining for comparison, check if fibMm1 is 1. If Yes, compare x with that remaining element. If match, return index.

```
# Fibonacci search
from bisect import bisect_left

# Returns index of x if present, else returns -1

def fibMonaccianSearch(arr, x, n):
    # write code here
    …

# Driver Code
arr = [10, 22, 35, 40, 45, 50, 80, 82, 85, 90, 100,235]
n = len(arr)
x = 235
ind = fibMonaccianSearch(arr, x, n)
if ind>=0:
  print("Found at index:",ind)
else:
  print(x,"isn't present in the array");
```

3. Sorting

3.1 Bubble Sort

Bubble Sort is the simplest sorting algorithm that works by repeatedly swapping the adjacent elements if they are in the wrong order. This algorithm is not suitable for large data sets as its average and worst-case time complexity is quite high.

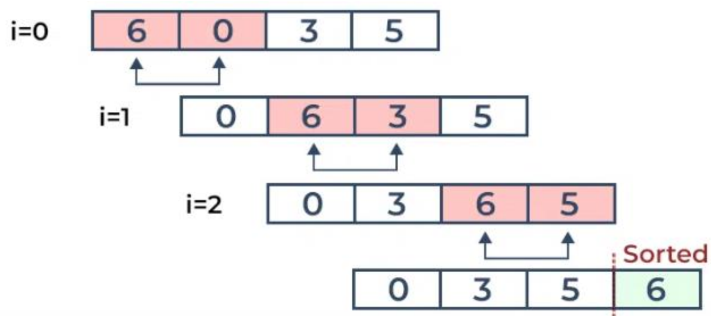
Bubble Sort Procedure:

1. Traverse from left and compare adjacent elements and the higher one is placed at right side.
2. In this way, the largest element is moved to the rightmost end at first.
3. This process is then continued to find the second largest and place it and so on until the data is sorted.

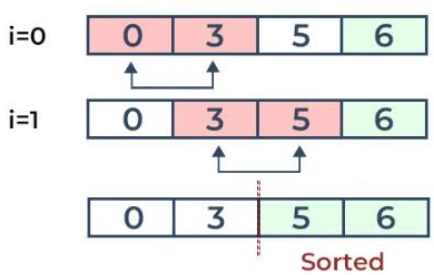
Input: arr = [6, 3, 0, 5]

Output:

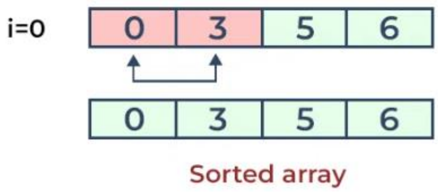
First Pass:



Second Pass:



Third Pass:




```
# Implementation of Bubble Sort

def bubbleSort(arr):
      # write code here
      …

# Driver code to test above
arr = [64, 34, 25, 12, 22, 11, 90]

bubbleSort(arr)

print("Sorted array:")
for i in range(len(arr)):
    print("%d" % arr[i], end=" ")
```

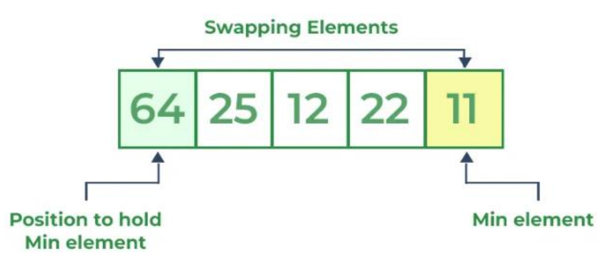
3.2 Selection Sort

Selection sort is a simple and efficient sorting algorithm that works by repeatedly selecting the smallest (or largest) element from the unsorted portion of the list and moving it to the sorted portion of the list. The algorithm repeatedly selects the smallest (or largest) element from the unsorted portion of the list and swaps it with the first element of the unsorted part. This process is repeated for the remaining unsorted portion until the entire list is sorted.

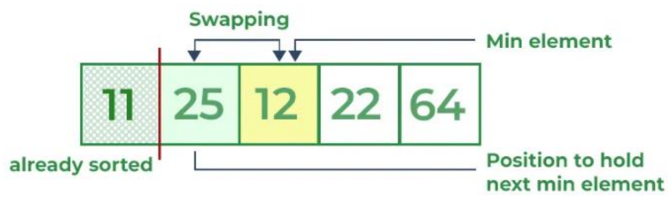
Input: arr = [64, 25, 12, 22, 11]

Output: arr = [11, 12, 22, 25, 64]

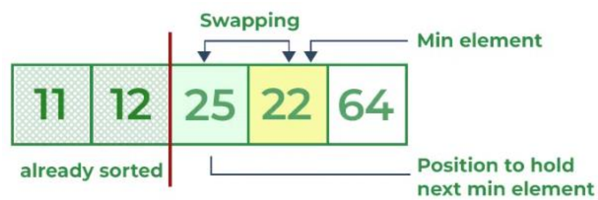
First Pass: For the first position in the sorted array, the whole array is traversed from index 0 to 4 sequentially. The first position where 64 is stored presently, after traversing whole array it is clear that 11 is the lowest value. Thus, replace 64 with 11. After one iteration 11, which happens to be the least value in the array, tends to appear in the first position of the sorted list.



Second Pass: For the second position, where 25 is present, again traverse the rest of the array in a sequential manner. After traversing, we found that 12 is the second lowest value in the array and it should appear at the second place in the array, thus swap these values.



Third Pass: Now, for third place, where 25 is present again traverse the rest of the array and find the third least value present in the array. While traversing, 22 came out to be the third least value and it should appear at the third place in the array, thus swap 22 with element present at third position.



Fourth Pass: Similarly, for fourth position traverse the rest of the array and find the fourth least element in the array. As 25 is the 4th lowest value hence, it will place at the fourth position.

Fifth Pass: At last the largest value present in the array automatically get placed at the last position in the array. The resulted array is the sorted array.

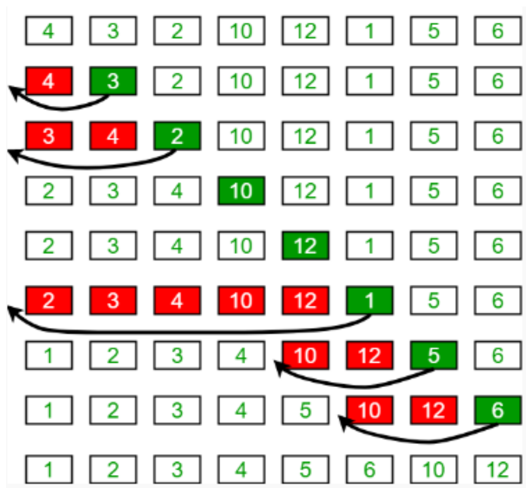

```
# Implementation of selection sort
import sys
A = [64, 25, 12, 22, 11]

# Traverse through all array elements
for i in range(len(A)):
      # write code here
       …
# Driver code
print ("Sorted array")
for i in range(len(A)):
    print("%d" %A[i],end=" , ")
```

3.3 Insertion Sort

Insertion sort is a simple sorting algorithm that works similar to the way you sort playing cards in your hands. The array is virtually split into a sorted and an unsorted part. Values from the unsorted part are picked and placed at the correct position in the sorted part.

Insertion Sort Procedure:

1. To sort an array of size N in ascending order iterate over the array and compare the current element (key) to its predecessor, if the key element is smaller than its predecessor, compare it to the elements before.
2. Move the greater elements one position up to make space for the swapped element.



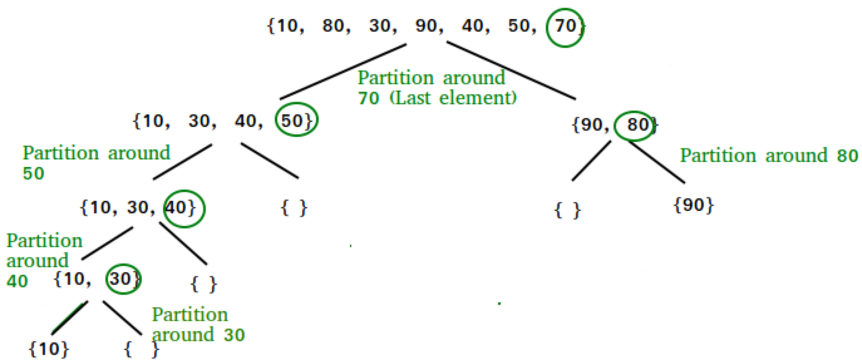
Input: arr = [4, 3, 2, 10, 12, 1, 5, 6]
Output: arr = [1, 2, 3, 4, 5, 6, 10, 12]

```
# Implementation of Insertion Sort

# Function to do insertion sort
def insertionSort(arr):
    # write code here
    …

# Driver code
arr = [12, 11, 13, 5, 6]
insertionSort(arr)
for i in range(len(arr)):
    print ("% d" % arr[i])
```

4. Divide and Conquer

4.1 Quick Sort

QuickSort is a sorting algorithm based on the Divide and Conquer algorithm that picks an element as a pivot and partitions the given array around the picked pivot by placing the pivot in its correct position in the sorted array. The key process in quickSort is a partition(). The target of partitions is to place the pivot (any element can be chosen to be a pivot) at its correct position in the sorted array and put all smaller elements to the left of the pivot, and all greater elements to the right of the pivot. Partition is done recursively on each side of the pivot after the pivot is placed in its correct position and this finally sorts the array.



The quick sort method can be summarized in three steps:

1. **Pick:** Select a pivot element.
2. **Divide:** Split the problem set, move smaller parts to the left of the pivot and larger items to the right.
3. **Repeat and combine:** Repeat the steps and combine the arrays that have previously been sorted.

Algorithm for Quick Sort Function:

```
//start –> Starting index,  end --> Ending index
Quicksort(array, start, end)
{
        if (start < end)
        {
                pIndex = Partition(A, start, end)
                Quicksort(A,start,pIndex-1)
                Quicksort(A,pIndex+1, end)
        }
}
```

Algorithm for Partition Function:

```
partition (array, start, end)
{
        // Setting rightmost Index as pivot
        pivot = arr[end];

        i = (start - 1)  // Index of smaller element and indicates the
            // right position of pivot found so far
        for (j = start; j <= end- 1; j++)
        {
                // If current element is smaller than the pivot
                if (arr[j] < pivot)
                {
                        i++;    // increment index of smaller element
                        swap arr[i] and arr[j]
                }
        }
```

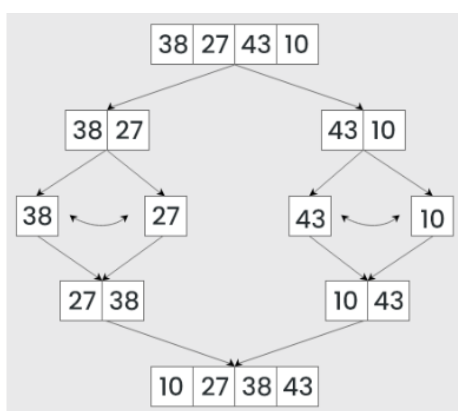
swap arr[i + 1] and arr[end])
        return (i + 1)
}

Input: arr = [10, 80, 30, 90, 40, 50, 70]
Output: arr = [10, 30, 40, 50, 70, 80, 90]

```
# Implementation of QuickSort

# Function to find the partition position
def partition(array, low, high):
    # write code here
    …

# Function to perform quicksort
def quicksort(array, low, high):
    # write code here
    …

# Driver code
array = [10, 7, 8, 9, 1, 5]
N = len(array)

# Function call
quicksort(array, 0, N - 1)
print('Sorted array:')
for x in array:
    print(x, end=" ")
```

4.2 Merge Sort

Merge sort is defined as a sorting algorithm that works by dividing an array into smaller subarrays, sorting each subarray, and then merging the sorted subarrays back together to form the final sorted array. In simple terms, we can say that the process of merge sort is to divide the array into two halves, sort each half, and then merge the sorted halves back together. This process is repeated until the entire array is sorted.



Input: arr = [12, 11, 13, 5, 6, 7]
Output: arr = [5, 6, 7, 11, 12, 13]

```
# Implementation of MergeSort

def mergeSort(arr):
    # write code here
    …

# print the list
def printList(arr):
    for i in range(len(arr)):
        print(arr[i], end=" ")
    print()


# Driver Code
arr = [12, 11, 13, 5, 6, 7]
print("Given array is")
printList(arr)
mergeSort(arr)
print("\nSorted array is ")
printList(arr)
```

4.3 Heap Sort

Heap sort is a comparison-based sorting technique based on Binary Heap data structure. It is similar to the selection sort where we first find the minimum element and place the minimum element at the beginning. Repeat the same process for the remaining elements.

Heap Sort Procedure:

First convert the array into heap data structure using heapify, then one by one delete the root node of the Max-heap and replace it with the last node in the heap and then heapify the root of the heap. Repeat this process until size of heap is greater than 1.

- Build a heap from the given input array.
- Repeat the following steps until the heap contains only one element:
    - Swap the root element of the heap (which is the largest element) with the last element of the heap.
    - Remove the last element of the heap (which is now in the correct position).
    - Heapify the remaining elements of the heap.
    - The sorted array is obtained by reversing the order of the elements in the input array.

Input: arr = [12, 11, 13, 5, 6, 7]
Output: Sorted array is 5 6 7 11 12 13

```
# Implementation of heap Sort
# To heapify subtree rooted at index i.
# n is size of heap
```

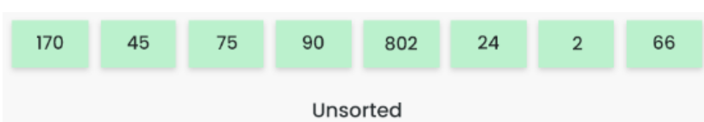
```python
def heapify(arr, N, i):
    # write code here
    …

# The main function to sort an array of given size
def heapSort(arr):
    # write code here
    …

# Driver code
arr = [12, 11, 13, 5, 6, 7]

# Function call
heapSort(arr)
N = len(arr)
print("Sorted array is")
for i in range(N):
    print("%d" % arr[i], end=" ")
```

4.4 Radix Sort

Radix Sort is a linear sorting algorithm that sorts elements by processing them digit by digit. It is an efficient sorting algorithm for integers or strings with fixed-size keys. Rather than comparing elements directly, Radix Sort distributes the elements into buckets based on each digit's value. By repeatedly sorting the elements by their significant digits, from the least significant to the most significant, Radix Sort achieves the final sorted order.

Radix Sort Procedure:
The key idea behind Radix Sort is to exploit the concept of place value.
1. It assumes that sorting numbers digit by digit will eventually result in a fully sorted list.
2. Radix Sort can be performed using different variations, such as Least Significant Digit (LSD) Radix Sort or Most Significant Digit (MSD) Radix Sort.

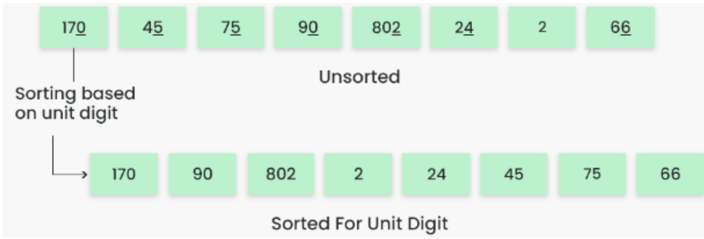
To perform radix sort on the array [170, 45, 75, 90, 802, 24, 2, 66], we follow these steps:



Step 1: Find the largest element in the array, which is 802. It has three digits, so we will iterate three times, once for each significant place.

Step 2: Sort the elements based on the unit place digits (X=0). We use a stable sorting technique, such as counting sort, to sort the digits at each significant place.

Sorting based on the unit place:
Perform counting sort on the array based on the unit place digits.
The sorted array based on the unit place is [170, 90, 802, 2, 24, 45, 75, 66]

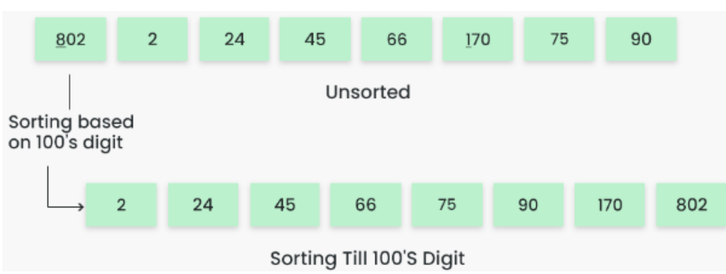


Step 3: Sort the elements based on the tens place digits.

Sorting based on the tens place:
Perform counting sort on the array based on the tens place digits.
The sorted array based on the tens place is [802, 2, 24, 45, 66, 170, 75, 90]

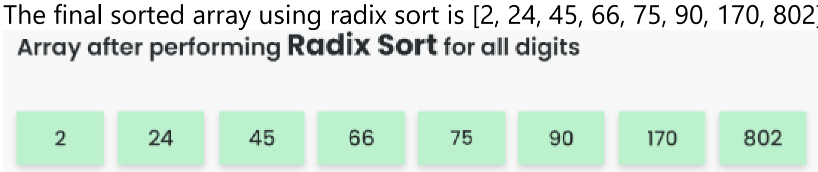
Step 4: Sort the elements based on the hundreds place digits.

Sorting based on the hundreds place:
Perform counting sort on the array based on the hundreds place digits.
The sorted array based on the hundreds place is [2, 24, 45, 66, 75, 90, 170, 802]



Step 5: The array is now sorted in ascending order.

The final sorted array using radix sort is [2, 24, 45, 66, 75, 90, 170, 802]




```
# Implementation of Radix Sort
# A function to do counting sort of arr[] according to the digit represented by exp.

def countingSort(arr, exp1):
    # write code here
    …
```

```
# Method to do Radix Sort
def radixSort(arr):
    # write code here
    …

# Driver code
arr = [170, 45, 75, 90, 802, 24, 2, 66]

# Function Call
radixSort(arr)

for i in range(len(arr)):
    print(arr[i],end=" ")
```

4.5 Shell Sort

Shell sort is mainly a variation of Insertion Sort. In insertion sort, we move elements only one position ahead. When an element has to be moved far ahead, many movements are involved. The idea of ShellSort is to allow the exchange of far items. In Shell sort, we make the array h-sorted for a large value of h. We keep reducing the value of h until it becomes 1. An array is said to be h-sorted if all sublists of every h'th element are sorted.

Shell Sort Procedure:

1. Initialize the value of gap size h
2. Divide the list into smaller sub-part. Each must have equal intervals to h
3. Sort these sub-lists using insertion sort
4. Repeat this step 1 until the list is sorted.
5. Print a sorted list.

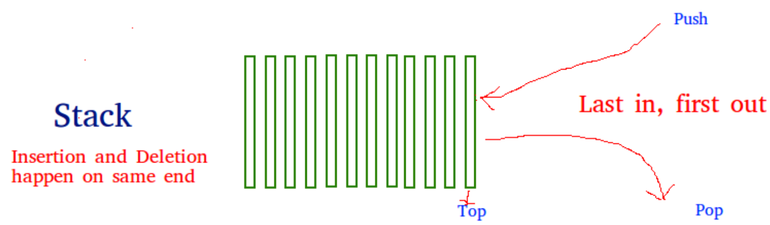
```
Procedure Shell_Sort(Array, N)
  While Gap < Length(Array) /3 :
            Gap = ( Interval * 3 ) + 1
  End While Loop
  While Gap > 0 :
    For (Outer = Gap; Outer < Length(Array); Outer++):
        Insertion_Value = Array[Outer]
            Inner = Outer;
        While Inner > Gap-1 And Array[Inner – Gap] >= Insertion_Value:
            Array[Inner] = Array[Inner – Gap]
            Inner = Inner – Gap
         End While Loop
           Array[Inner] = Insertion_Value
    End For Loop
    Gap = (Gap -1) /3;
  End While Loop
End Shell_Sort
```

```
# Implementation of Shell Sort

def shellSort(arr, n):
    # write code here
    …

# Driver code
arr = [12, 34, 54, 2, 3]
print("input array:",arr)

shellSort(arr,len(arr))
print("sorted array",arr)
```

5. Stack

5.1 Stack implementation using List

A stack is a linear data structure that stores items in a Last-In/First-Out (LIFO) or First-In/Last-Out (FILO) manner. In stack, a new element is added at one end and an element is removed from that end only. The insert and delete operations are often called push and pop.



The functions associated with stack are:

- **empty()** – Returns whether the stack is empty
- **size()** – Returns the size of the stack
- **top() / peek()** – Returns a reference to the topmost element of the stack
- **push(a)** – Inserts the element 'a' at the top of the stack
- **pop()** – Deletes the topmost element of the stack

```
# Stack implementation using list
top=0
mymax=5
def createStack():
    stack=[]
    return stack
def isEmpty(stack):
    # write code here
    …
def Push(stack,item):
    # write code here
    …
```

```
def Pop(stack):
    # write code here
    …
# create a stack object
stack = createStack()
while True:
    print("1.Push")
    print("2.Pop")
    print("3.Display")
    print("4.Quit")
    # write code here
    …
```

5.2 Balanced Parenthesis Checking

Given an expression string, write a python program to find whether a given string has balanced parentheses or not.

Input: {[]{()}}

Output: Balanced

Input: [{}{}(]

Output: Unbalanced

Using stack One approach to check balanced parentheses is to use stack. Each time, when an open parentheses is encountered push it in the stack, and when closed parenthesis is encountered, match it with the top of stack and pop it. If stack is empty at the end, return Balanced otherwise, Unbalanced.

```
# Check for balanced parentheses in an expression
open_list = ["[","{","("]
close_list = ["]","}",")"]

# Function to check parentheses
def check(myStr):
    # write code here
    …
```

5.3 Evaluation of Postfix Expression

Given a postfix expression, the task is to evaluate the postfix expression. Postfix expression: The expression of the form "a b operator" (ab+) i.e., when a pair of operands is followed by an operator.

Input: str = "2 3 1 * + 9 -"

Output: -4

Explanation: If the expression is converted into an infix expression, it will be 2 + (3 * 1) – 9 = 5 – 9 = -4.

Input: str = "100 200 + 2 / 5 * 7 +"

Output: 757

Procedure for evaluation postfix expression using stack:

- Create a stack to store operands (or values).
- Scan the given expression from left to right and do the following for every scanned element.
    - If the element is a number, push it into the stack.
    - If the element is an operator, pop operands for the operator from the stack. Evaluate the operator and push the result back to the stack.
- When the expression is ended, the number in the stack is the final answer.

```
# Evaluate value of a postfix expression

# Class to convert the expression
class Evaluate:

    # Constructor to initialize the class variables
    def __init__(self, capacity):
        self.top = -1
        self.capacity = capacity

        # This array is used a stack
        self.array = []

    # Check if the stack is empty
    def isEmpty(self):
        # write code here

        …

    def peek(self):
        # write code here

        …

    def pop(self):
        # write code here

        …

    def push(self, op):
        # write code here

        …

    def evaluatePostfix(self, exp):
        # write code here

        …

# Driver code
exp = "231*+9-"
obj = Evaluate(len(exp))

# Function call
print("postfix evaluation: %d" % (obj.evaluatePostfix(exp)))
```

5.4 Infix to Postfix Expression Conversion

For a given Infix expression, convert it into Postfix form.

Infix expression: The expression of the form "a operator b" (a + b) i.e., when an operator is in-between every pair of operands.
Postfix expression: The expression of the form "a b operator" (ab+) i.e., When every pair of operands is followed by an operator.

Infix to postfix expression conversion procedure:

1. Scan the infix expression from left to right.

2. If the scanned character is an operand, put it in the postfix expression.

3. Otherwise, do the following

   - If the precedence and associativity of the scanned operator are greater than the precedence and associativity of the operator in the stack [or the stack is empty or the stack contains a '('], then push it in the stack. ['^' operator is right associative and other operators like '+','–','*' and '/' are left-associative].

   - Check especially for a condition when the operator at the top of the stack and the scanned operator both are '^'. In this condition, the precedence of the scanned operator is higher due to its right associativity. So it will be pushed into the operator stack.

   - In all the other cases when the top of the operator stack is the same as the scanned operator, then pop the operator from the stack because of left associativity due to which the scanned operator has less precedence.

   - Else, Pop all the operators from the stack which are greater than or equal to in precedence than that of the scanned operator.

   - After doing that Push the scanned operator to the stack. (If you encounter parenthesis while popping then stop there and push the scanned operator in the stack.)

4. If the scanned character is a '(', push it to the stack.

5. If the scanned character is a ')', pop the stack and output it until a '(' is encountered, and discard both the parenthesis.

6. Repeat steps 2-5 until the infix expression is scanned.

7. Once the scanning is over, Pop the stack and add the operators in the postfix expression until it is not empty.

8. Finally, print the postfix expression.

Input: A + B * C + D
Output: A B C * + D +

Input: ((A + B) – C * (D / E)) + F
Output: A B + C D E / * - F +

```python
# Convert infix expression to postfix
# Class to convert the expression

class Conversion:

    # Constructor to initialize the class variables
    def __init__(self, capacity):
        self.top = -1
        self.capacity = capacity

        # This array is used a stack
        self.array = []


        # Precedence setting
        self.output = []
        self.precedence = {'+': 1, '-': 1, '*': 2, '/': 2, '^': 3}

    # Check if the stack is empty
    def isEmpty(self):
        # write code here
        …

    # Return the value of the top of the stack
    def peek(self):
        # write code here
        …

    # Pop the element from the stack
    def pop(self):
        # write code here
        …

    # Push the element to the stack
    def push(self, op):
        # write code here
        …

    # A utility function to check is the given character is operand
    def isOperand(self, ch):
        # write code here
        …

    # Check if the precedence of operator is strictly less than top of stack or not
    def notGreater(self, i):
        # write code here
        …

    # The main function that converts given infix expression
```

```
    # to postfix expression
    def infixToPostfix(self, exp):
      # write code here
        …

# Driver code
exp = "a+b*(c^d-e)^(f+g*h)-i"
obj = Conversion(len(exp))

# Function call
obj.infixToPostfix(exp)
```

5.5 Reverse a Stack

The stack is a linear data structure which works on the LIFO concept. LIFO stands for last in first out. In the stack, the insertion and deletion are possible at one end the end is called the top of the stack. Define two recursive functions BottomInsertion() and Reverse() to reverse a stack using Python. Define some basic function of the stack like push(), pop(), show(), empty(), for basic operation like respectively append an item in stack, remove an item in stack, display the stack, check the given stack is empty or not.

BottomInsertion(): this method append element at the bottom of the stack and BottomInsertion accept two values as an argument first is stack and the second is elements, this is a recursive method.

Reverse(): the method is reverse elements of the stack, this method accept stack as an argument Reverse() is also a Recursive() function. Reverse() is invoked BottomInsertion() method for completing the reverse operation on the stack.

Input: Elements = [1, 2, 3, 4, 5]
Output: Original Stack
5
4
3
2
1
Stack after Reversing
1
2
3
4
5

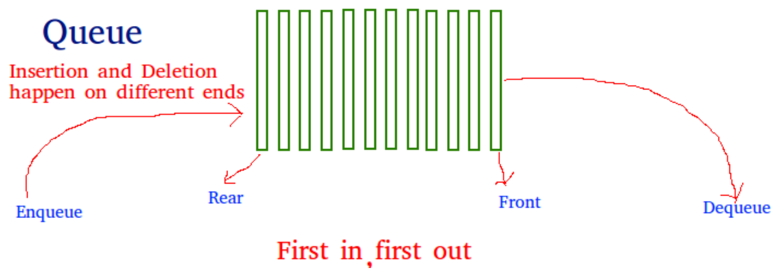
```
# create class for stack
class Stack:

    # create empty list
    def __init__(self):
        self.Elements = []

    # push() for insert an element
    def push(self, value):
        self.Elements.append(value)
```

```
    # pop() for remove an element
    def pop(self):
        return self.Elements.pop()

    # empty() check the stack is empty of not
    def empty(self):
        return self.Elements == []

    # show() display stack
    def show(self):
        for value in reversed(self.Elements):
            print(value)
 # Insert_Bottom() insert value at bottom
def BottomInsert(s, value):
   # write code here

   …

# Reverse() reverse the stack
def Reverse(s):
   # write code here

   …

# create object of stack class
stk = Stack()

stk.push(1)
stk.push(2)
stk.push(3)
stk.push(4)
stk.push(5)

print("Original Stack")
stk.show()

print("\nStack after Reversing")
Reverse(stk)
stk.show()
```

6. Queue

6.1 Linear Queue

Linear queue is a linear data structure that stores items in First in First out (FIFO) manner. With a queue the least recently added item is removed first. A good example of queue is any queue of consumers for a resource where the consumer that came first is served first.




```
# Static implementation of linear queue
front=0
rear=0
mymax=5
def createQueue():
    queue=[]   #empty list
    return queue
def isEmpty(queue):
    # write code here
    …
def enqueue(queue,item):  # insert an element into the queue
    # write code here
    …
def dequeue(queue):  #remove an element from the queue
    # write code here
    …
# Driver code
queue = createQueue()
while True:
    print("1.Enqueue")
    print("2.Dequeue")
    print("3.Display")
    print("4.Quit")
    # write code here
    …
```

6.2 Stack using Queues

Implement a last-in-first-out (LIFO) stack using only two queues. The implemented stack should support all the functions of a normal stack (push, top, pop, and empty).

- void push(int x) Pushes element x to the top of the stack.
- int pop() Removes the element on the top of the stack and returns it.

- int top() Returns the element on the top of the stack.
- boolean empty() Returns true if the stack is empty, false otherwise.

Input:

["MyStack", "push", "push", "top", "pop", "empty"]

[[], [1], [2], [], [], []]

Output:

[null, null, null, 2, 2, false]

```
class MyStack:

    def __init__(self):
       # write code here
       …
    def push(self, x: int) -> None:
       # write code here
       …
    def pop(self) -> int:
       # write code here
       …
    def top(self) -> int:
       # write code here
       …
    def empty(self) -> bool:
       # write code here
       …
# Your MyStack object will be instantiated and called as such:
# obj = MyStack()
# obj.push(x)
# param_2 = obj.pop()
# param_3 = obj.top()
# param_4 = obj.empty()
```

6.3 Queue using Stacks

Implement a first in first out (FIFO) queue using only two stacks. The implemented queue should support all the functions of a normal queue (push, peek, pop, and empty).

- void push(int x) Pushes element x to the back of the queue.
- int pop() Removes the element from the front of the queue and returns it.
- int peek() Returns the element at the front of the queue.
- boolean empty() Returns true if the queue is empty, false otherwise.

Input:

["MyQueue", "push", "push", "peek", "pop", "empty"]

[[], [1], [2], [], [], []]

Output:

[null, null, null, 1, 1, false]

```
class MyQueue:

    def __init__(self):
       # write code here
       …
    def push(self, x: int) -> None:
       # write code here
       …
    def pop(self) -> int:
       # write code here
       …
    def peek(self) -> int:
       # write code here
       …
    def empty(self) -> bool:
       # write code here
       …
# Your MyQueue object will be instantiated and called as such:
# obj = MyQueue()
# obj.push(x)
# param_2 = obj.pop()
# param_3 = obj.peek()
# param_4 = obj.empty()
```

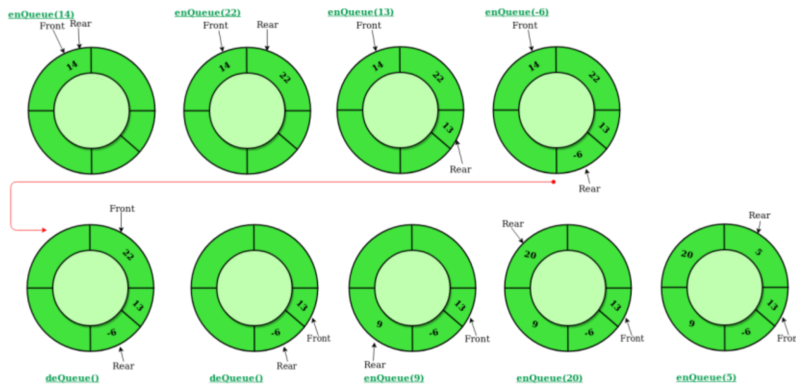
6.4 Circular Queue

A Circular Queue is an extended version of a normal queue where the last element of the queue is connected to the first element of the queue forming a circle. The operations are performed based on FIFO (First In First Out) principle. It is also called 'Ring Buffer'.

Operations on Circular Queue:

- **Front:** Get the front item from the queue.
- **Rear:** Get the last item from the queue.
- **enQueue(value)** This function is used to insert an element into the circular queue. In a circular queue, the new element is always inserted at the rear position.
    - Check whether the queue is full – [i.e., the rear end is in just before the front end in a circular manner].

- If it is full then display Queue is full.
    - If the queue is not full then, insert an element at the end of the queue.
- **deQueue()** This function is used to delete an element from the circular queue. In a circular queue, the element is always deleted from the front position.
  - Check whether the queue is Empty.
  - If it is empty then display Queue is empty.
    - If the queue is not empty, then get the last element and remove it from the queue.



Implement Circular Queue using Array:

1. Initialize an array queue of size **n**, where n is the maximum number of elements that the queue can hold.
2. Initialize two variables front and rear to -1.
3. **Enqueue:** To enqueue an element **x** into the queue, do the following:
   - Increment rear by 1.
   - If **rear** is equal to n, set **rear** to 0.
   - If **front** is -1, set **front** to 0.
   - Set queue[rear] to x.
4. **Dequeue:** To dequeue an element from the queue, do the following:
   - Check if the queue is empty by checking if **front** is -1.
   - If it is, return an error message indicating that the queue is empty.
   - Set **x** to queue [front].
   - If **front** is equal to **rear**, set **front** and **rear** to -1.

- Otherwise, increment **front** by 1 and if **front** is equal to n, set **front** to 0.
- Return x.

```
class CircularQueue():

    # constructor
    def __init__(self, size): # initializing the class
        self.size = size

        # initializing queue with none
        self.queue = [None for i in range(size)]
        self.front = self.rear = -1

    def enqueue(self, data):
         # Write code here
         …

    def dequeue(self):
         # Write code here
         …

    def display(self):
        # Write code here
        …
# Driver Code
ob = CircularQueue(5)
ob.enqueue(14)
ob.enqueue(22)
ob.enqueue(13)
ob.enqueue(-6)
ob.display()
print ("Deleted value = ", ob.dequeue())
print ("Deleted value = ", ob.dequeue())
ob.display()
ob.enqueue(9)
ob.enqueue(20)
ob.enqueue(5)
ob.display()
```

6.5 Deque (Doubly Ended Queue)

In a Deque (Doubly Ended Queue), one can perform insert (append) and delete (pop) operations from both the ends of the container. There are two types of Deque:

1. **Input Restricted Deque:** Input is limited at one end while deletion is permitted at both ends.
2. **Output Restricted Deque:** Output is limited at one end but insertion is permitted at both ends.

Operations on Deque:

1. **append():** This function is used to insert the value in its argument to the right end of the deque.
2. **appendleft():** This function is used to insert the value in its argument to the left end of the deque.
3. **pop():** This function is used to delete an argument from the right end of the deque.

4. **popleft():** This function is used to delete an argument from the left end of the deque.
5. **index(ele, beg, end):** This function returns the first index of the value mentioned in arguments, starting searching from beg till end index.
6. **insert(i, a):** This function inserts the value mentioned in arguments(a) at index(i) specified in arguments.
7. **remove():** This function removes the first occurrence of the value mentioned in arguments.
8. **count():** This function counts the number of occurrences of value mentioned in arguments.
9. **len(dequeue):** Return the current size of the dequeue.
10. **Deque[0]:** We can access the front element of the deque using indexing with de[0].
11. **Deque[-1]:** We can access the back element of the deque using indexing with de[-1].
12. **extend(iterable):** This function is used to add multiple values at the right end of the deque. The argument passed is iterable.
13. **extendleft(iterable):** This function is used to add multiple values at the left end of the deque. The argument passed is iterable. Order is reversed as a result of left appends.
14. **reverse():** This function is used to reverse the order of deque elements.
15. **rotate():** This function rotates the deque by the number specified in arguments. If the number specified is negative, rotation occurs to the left. Else rotation is to right.

```
# importing "collections" for deque operations
import collections

# initializing deque
de = collections.deque([1, 2, 3])
print("deque: ", de)

# using append() to insert 4 at the end of deque
# Write code here

# Printing modified deque
# Write code here

# using appendleft() to insert 6 at the beginning of deque
# Write code here


# Printing modified deque
# Write code here

# using pop() to delete 4 from the right end of deque
# Write code here

# Printing modified deque
# Write code here

# using popleft() to delete 6 from the left end of deque
# Write code here
```

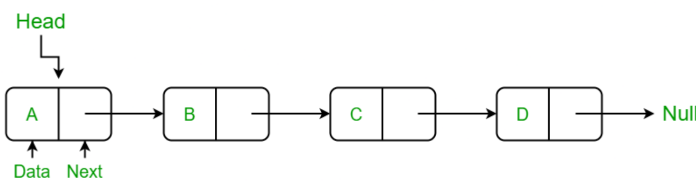
```python
# Printing modified deque
# Write code here

# using insert() to insert the value 3 at 5th position
# Write code here

# printing modified deque
# Write code here

# using count() to count the occurrences of 3
# Write code here

# using remove() to remove the first occurrence of 3
# Write code here

# Printing modified deque
# Write code here

# Printing current size of deque
# Write code here

# using pop() to delete 6 from the right end of deque
# Write code here

# Printing modified deque
# Write code here

# Printing current size of deque
# Write code here

# Accessing the front element of the deque
# Write code here

# Accessing the back element of the deque
# Write code here

# using extend() to add 4,5,6 to right end
# Write code here

# Printing modified deque
# Write code here

# using extendleft() to add 7,8,9 to left end
# Write code here

# Printing modified deque
# Write code here

# using rotate() to rotate the deque rotates by 3 to left
```

```
# Write code here

# Printing modified deque
# Write code here

# using reverse() to reverse the deque
# Write code here

# Printing modified deque
# Write code here
```

7. Linked List

7.1 Singly Linked List

A singly linked list is a linear data structure in which the elements are not stored in contiguous memory locations and each element is connected only to its next element using a pointer.



Creating a linked list involves the following operations:

1. Creating a Node class:
2. Insertion at beginning:
3. Insertion at end
4. Insertion at middle
5. Update the node
6. Deletion at beginning
7. Deletion at end
8. Deletion at middle
9. Remove last node
10. Linked list traversal
11. Get length

```
# Create a Node class to create a node
class Node:
    def __init__(self, data):
        self.data = data
        self.next = None


# Create a LinkedList class
class LinkedList:
```

```
    def __init__(self):
        self.head = None

    # Method to add a node at begin of LL
    def insertAtBegin(self, data):
        # Write code here
        …
    # Method to add a node at any index, Indexing starts from 0.
    def insertAtIndex(self, data, index):
        # Write code here
        …
     # Method to add a node at the end of LL
    def insertAtEnd(self, data):
        # Write code here
        …
     # Update node of a linked list at given position
    def updateNode(self, val, index):
        # Write code here
        …
    # Method to remove first node of linked list
    def remove_first_node(self):
        # Write code here
        …
    # Method to remove last node of linked list
    def remove_last_node(self):
        # Write code here
        …
    # Method to remove at given index
    def remove_at_index(self, index):
        # Write code here
        …
     # Method to remove a node from linked list
    def remove_node(self, data):
        # Write code here
        …
     # Print the size of linked list
    def sizeOfLL(self):
        # Write code here
        …
```

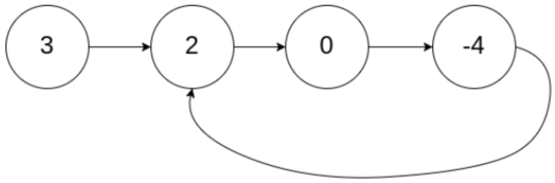
```
    # print method for the linked list
    def printLL(self):
        # Write code here
        …
# create a new linked list
llist = LinkedList()

# add nodes to the linked list
llist.insertAtEnd('a')
llist.insertAtEnd('b')
llist.insertAtBegin('c')
llist.insertAtEnd('d')
llist.insertAtIndex('g', 2)
# print the linked list
print("Node Data")
llist.printLL()
# remove a nodes from the linked list
print("\nRemove First Node")
llist.remove_first_node()
print("Remove Last Node")
llist.remove_last_node()
print("Remove Node at Index 1")
llist.remove_at_index(1)
# print the linked list again
print("\nLinked list after removing a node:")
llist.printLL()
print("\nUpdate node Value")
llist.updateNode('z', 0)
llist.printLL()
print("\nSize of linked list :", end=" ")
print(llist.sizeOfLL())
```

7.2 Linked List Cycle

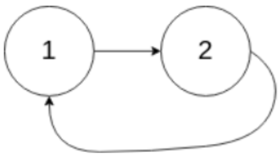
Given head, the head of a linked list, determine if the linked list has a cycle in it. There is a cycle in a linked list if there is some node in the list that can be reached again by continuously following the next pointer. Internally, pos is used to denote the index of the node that tail's next pointer is connected to.

Note that pos is not passed as a parameter.

Return true if there is a cycle in the linked list. Otherwise, return false.



Input: head = [3, 2, 0, -4], pos = 1
Output: true
Explanation: There is a cycle in the linked list, where the tail connects to the 1st node (0-indexed).



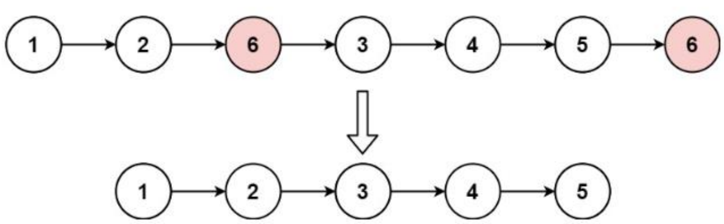
Input: head = [1, 2], pos = 0
Output: true
Explanation: There is a cycle in the linked list, where the tail connects to the 0th node.

Input: head = [1], pos = -1
Output: false
Explanation: There is no cycle in the linked list.

```
# Definition for singly-linked list.
class ListNode:
    def __init__(self, x):
        self.val = x
        self.next = None

class Solution:
    def hasCycle(self, head):
        # Write code here
        …
```

7.3 Remove Linked List Elements

Given the head of a linked list and an integer val, remove all the nodes of the linked list that has Node.val == val, and return the new head.



Input: head = [1, 2, 6, 3, 4, 5, 6], val = 6

Output: [1, 2, 3, 4, 5]

Input: head = [ ], val = 1

Output: [ ]

Input: head = [7, 7, 7, 7], val = 7

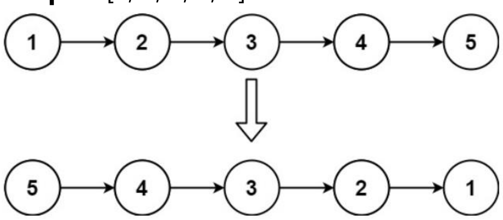
Output: [ ]

```
# Definition for singly-linked list.
class ListNode:
   def __init__(self, val=0, next=None):
       self.val = val
       self.next = next

class Solution:
    def removeElements(self, head, val):
        # Write code here
        …
```

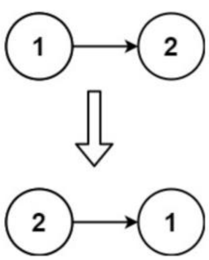
7.4 Reverse Linked List

Given the head of a singly linked list, reverse the list, and return the reversed list.

Input: head = [1, 2, 3, 4, 5]
Output: [5, 4, 3, 2, 1]



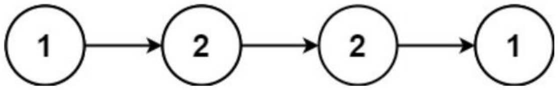
Input: head = [1, 2]
Output: [2, 1]




```
# Definition for singly-linked list.
class ListNode:
   def __init__(self, val=0, next=None):
       self.val = val
       self.next = next

class Solution:
    def reverseList(self, head):
        # Write code here
        …
```

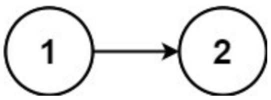
7.5 Palindrome Linked List

Given the head of a singly linked list, return true if it is a palindrome or false otherwise.



Input: head = [1, 2, 2, 1]

Output: true



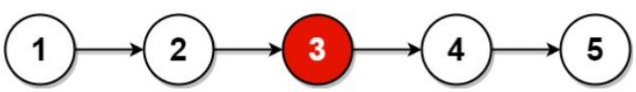
Input: head = [1, 2]

Output: false

```
# Definition for singly-linked list.
class ListNode:
   def __init__(self, val=0, next=None):
       self.val = val
       self.next = next

class Solution:
    def isPalindrome(self, head):
        # Write code here
         …
```

7.6 Middle of the Linked List

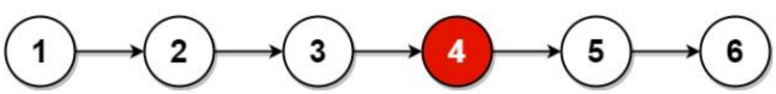
Given the head of a singly linked list, return the middle node of the linked list. If there are two middle nodes, return the second middle node.



Input: head = [1, 2, 3, 4, 5]

Output: [3, 4, 5]

Explanation: The middle node of the list is node 3.



Input: head = [1, 2, 3, 4, 5, 6]

Output: [4, 5, 6]

Explanation: Since the list has two middle nodes with values 3 and 4, we return the second one.

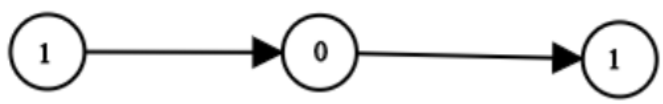
```
# Definition for singly-linked list.
class ListNode:
   def __init__(self, val=0, next=None):
       self.val = val
       self.next = next
```

```python
class Solution:
    def middleNode(self, head):
        # Write code here
        …
```

7.7 Convert Binary Number in a Linked List to Integer

Given head which is a reference node to a singly-linked list. The value of each node in the linked list is either 0 or 1. The linked list holds the binary representation of a number.

Return the decimal value of the number in the linked list. The most significant bit is at the head of the linked list.



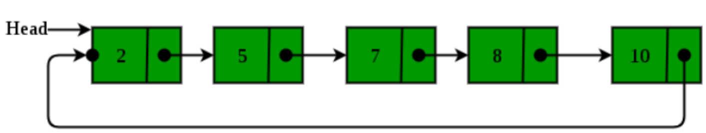
Input: head = [1, 0, 1]
Output: 5
Explanation: (101) in base 2 = (5) in base 10
Input: head = [0]
Output: 0

```python
# Definition for singly-linked list.
class ListNode:
   def __init__(self, val=0, next=None):
       self.val = val
       self.next = next

class Solution:
    def getDecimalValue(self, head):
        # Write code here
        …
```

8. Circular Single Linked List and Doubly Linked List

8.1 Circular Linked List

The circular linked list is a linked list where all nodes are connected to form a circle. In a circular linked list, the first node and the last node are connected to each other which forms a circle. There is no NULL at the end.



Operations on the circular linked list:

1. Insertion at the beginning
2. Insertion at the end
3. Insertion in between the nodes
4. Deletion at the beginning

5. Deletion at the end
6. Deletion in between the nodes
7. Traversal

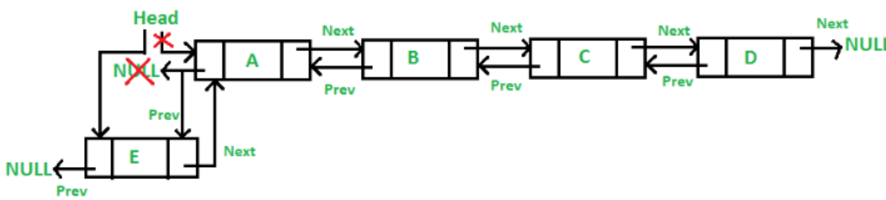
```
# Circular linked list operations

class Node:
    def __init__(self, data):
        self.data = data
        self.next = None

class CircularLinkedList:
    def __init__(self):
        self.last = None
    def addToEmpty(self, data):
       # Write code here
        …

    # add node to the front
    def addFront(self, data):
       # Write code here
        …

    # add node to the end
    def addEnd(self, data):
        # Write code here
        …

    # insert node after a specific node
    def addAfter(self, data, item):
       # Write code here
        …

    # delete a node
    def deleteNode(self, last, key):
       # Write code here
        …

    def traverse(self):
        # Write code here
        …
# Driver Code
cll = CircularLinkedList()

last = cll.addToEmpty(6)
last = cll.addEnd(8)
last = cll.addFront(2)
last = cll.addAfter(10, 2)

cll.traverse()
last = cll.deleteNode(last, 8)
print()
cll.traverse()
```

8.2 Doubly Linked List

The A doubly linked list is a type of linked list in which each node consists of 3 components:

1. *prev - address of the previous node
2. data - data item
3. *next - address of next node.

Double Linked List Node



Operations on the Double Linked List:

1. Insertion at the beginning
2. Insertion at the end
3. Insertion in between the nodes
4. Deletion at the beginning
5. Deletion at the end
6. Deletion in between the nodes
7. Traversal

```
# Implementation of doubly linked list
class Node:
    def __init__(self,data):
        self.data=data
        self.next=self.prev=None

class DLinkedList:
    def __init__(self):
        self.head=None
        self.ctr=0
    def insert_beg(self,data):
        # Write code here
        …
    def insert_end(self,data):
        # Write code here
        …
    def delete_beg(self):
        # Write code here
        …
    def delete_end(self):
        # Write code here
```

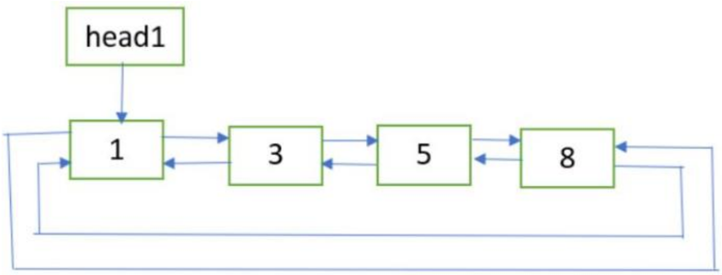
```
        …
    def insert_pos(self,pos,data):
        # Write code here
        …
    def delete_pos(self,pos):
        # Write code here
        …
    def traverse_f(self):
        # Write code here
        …
    def traverse_r(self):
        # Write code here
        …

def menu():
    print("1.Insert at beginning")
    print("2.Insert at position")
    print("3.Insert at end")
    print("4.Delete at beginning")
    print("5.Delete at position")
    print("6.Delete at end")
    print("7.Count no of nodes")
    print("8.Traverse forward")
    print("9.Traverse reverse")
    print("10.Quit")
    ch=eval(input("Enter choice:"))
    return ch

print("*****************Double linked list**************")
d=DLinkedList()
while True :
    ch=menu()
    if ch==1:
        data=eval(input("Enter data:"))
        d.insert_beg(data)

    elif ch==2:
        data=eval(input("Enter data:"))
        pos=int(input("Enter position:"))
        d.insert_pos(pos,data)

    elif ch==3:
        data=eval(input("Enter data:"))
        d.insert_end(data)

    elif ch==4:
        d.delete_beg()

    elif ch==5:
        pos=int(input("Enter position:"))
        d.delete_pos(pos)

    elif ch==6:
        d.delete_end()

    elif ch==7:
```

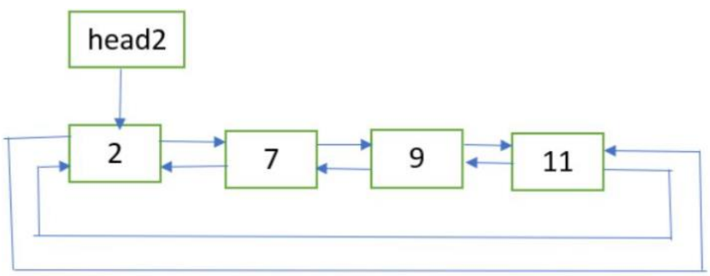
```
        print("Number of nodes",d.ctr)

    elif ch==8:
        d.traverse_f()
    elif ch==9:
        d.traverse_r()

    else:
        print("Exit")
        break
```

8.3 Sorted Merge of Two Sorted Doubly Circular Linked Lists

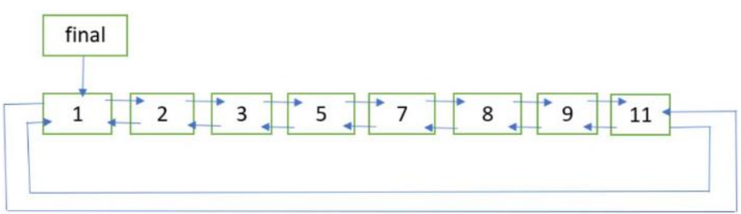
Given two sorted Doubly circular Linked List containing n1 and n2 nodes respectively. The problem is to merge the two lists such that resultant list is also in sorted order.

Input: List 1 and List 2





Output: Merged List



Procedure for Merging Doubly Linked List:

1. If head1 == NULL, return head2.

2. If head2 == NULL, return head1.

3. Let **last1** and **last2** be the last nodes of the two lists respectively. They can be obtained with the help of the previous links of the first nodes.

4. Get pointer to the node which will be the last node of the final list. If last1.data < last2.data, then **last_node** = last2, Else **last_node** = last1.

5. Update last1.next = last2.next = NULL.

6. Now merge the two lists as two sorted doubly linked list are being merged. Refer **merge** procedure of this post. Let the first node of the final list be **finalHead**.

7. Update finalHead.prev = last_node and last_node.next = finalHead.

8. Return **finalHead**.

```
# Implementation for Sorted merge of two sorted doubly circular linked list

import math

class Node:
    def __init__(self, data):
        self.data = data
        self.next = None
        self.prev = None

# A utility function to insert a new node at the beginning
# of doubly circular linked list
def insert(head_ref, data):
     # Write code here
     …
# function for Sorted merge of two sorted doubly linked list
def merge(first, second):
    # Write code here
    …
# function for Sorted merge of two sorted doubly circular linked list
def mergeUtil(head1, head2):
    # Write code here
    …
# function to print the list
def printList(head):
    # Write code here
    …
# Driver Code
head1 = None
head2 = None
# list 1:
head1 = insert(head1, 8)
head1 = insert(head1, 5)
head1 = insert(head1, 3)
head1 = insert(head1, 1)
# list 2:
head2 = insert(head2, 11)
head2 = insert(head2, 9)
```

```
head2 = insert(head2, 7)
head2 = insert(head2, 2)

newHead = mergeUtil(head1, head2)

print("Final Sorted List: ", end = "")
printList(newHead)
```

8.4 Delete all occurrences of a given key in a Doubly Linked List

Given a doubly linked list and a key x. The problem is to delete all occurrences of the given key x from the doubly linked list.

Input: 2 <-> 2 <-> 10 <-> 8 <-> 4 <-> 2 <-> 5 <-> 2
x = 2
Output: 10 <-> 8 <-> 4 <-> 5

Algorithm:
delAllOccurOfGivenKey (head_ref, x)
  if head_ref == NULL
    return
  Initialize **current** = head_ref
  Declare **next**
  while current != NULL
    if current->data == x
      next = current->next
      deleteNode(head_ref, current)
      current = next
    else
      current = current->next

```
# Implementation to delete all occurrences of a given key in a doubly linked list
import math

# a node of the doubly linked list
class Node:
    def __init__(self,data):
        self.data = data
        self.next = None
        self.prev = None

# Function to delete a node in a Doubly Linked List.
# head_ref --> pointer to head node pointer.
# del --> pointer to node to be deleted.
def deleteNode(head, delete):
    # Write code here
    …

# function to delete all occurrences of the given key 'x'
def deleteAllOccurOfX(head, x):
    # Write code here
    …

# Function to insert a node at the beginning of the Doubly Linked List
```

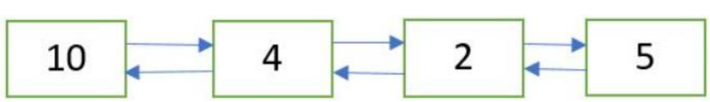
```
def push(head,new_data):
    # Write code here
    …

# Function to print nodes in a given doubly linked list
def printList(head):
    # Write code here
    …

# Driver Code
# Start with the empty list
head = None
# Create the doubly linked list:
head = push(head, 2)
head = push(head, 5)
head = push(head, 2)
head = push(head, 4)
head = push(head, 8)
head = push(head, 10)
head = push(head, 2)
head = push(head, 2)
print("Original Doubly linked list:")
printList(head)
x = 2
# delete all occurrences of 'x'
head = deleteAllOccurOfX(head, x)
print("\nDoubly linked list after deletion of ",x,":")
printList(head)
```

8.5 Delete a Doubly Linked List Node at a Given Position

Given a doubly linked list and a position n. The task is to delete the node at the given position n from the beginning.

Input: Initial doubly linked list

Output: Doubly Linked List after deletion of node at position n = 2



Procedure:

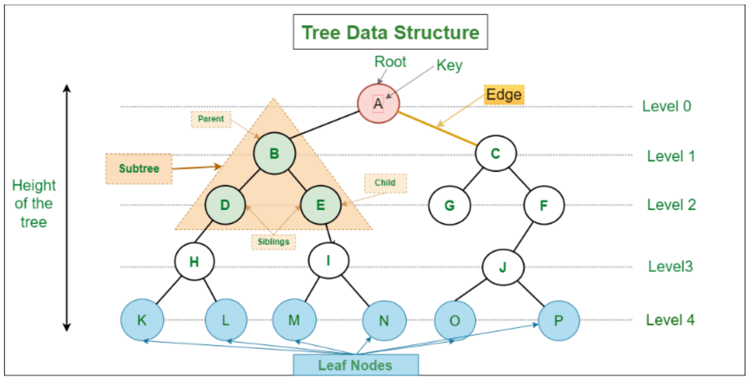
1. Get the pointer to the node at position n by traversing the doubly linked list up to the nth node from the beginning.
2. Delete the node using the pointer obtained in Step 1.

```
# Python implementation to delete a doubly Linked List node
# at the given position

# A node of the doubly linked list
class Node:

    # Constructor to create a new node
    def __init__(self, data):
        self.data = data
        self.next = None
        self.prev = None

# Function to delete a node in a Doubly Linked List.
# head_ref -. pointer to head node pointer.
# del -. pointer to node to be deleted.
def deleteNode(head_ref, del_):
    # Write code here
    …

# Function to delete the node at the given position
# in the doubly linked list
def deleteNodeAtGivenPos(head_ref, n):
    # Write code here
    …

# Function to insert a node at the beginning of the Doubly Linked List
def push(head_ref, new_data):
    # Write code here
    …
 # Function to print nodes in a given doubly linked list
def printList(head):
    # Write code here
    …
# Driver Code
# Start with the empty list
head = None
head = push(head, 5)
head = push(head, 2)
head = push(head, 4)
head = push(head, 8)
head = push(head, 10)
print("Doubly linked list before deletion:")
printList(head)

n = 2
 # delete node at the given position 'n'
head = deleteNodeAtGivenPos(head, n)
print("\nDoubly linked list after deletion:")
printList(head)
```

9. Trees

9.1 Tree Creation and Basic Tree Terminologies

A tree data structure is a hierarchical structure that is used to represent and organize data in a way that is easy to navigate and search. It is a collection of nodes that are connected by edges and has a hierarchical relationship between the nodes.



Basic Terminologies in Tree:

1. **Parent Node:** The node which is a predecessor of a node is called the parent node of that node. {B} is the parent node of {D, E}.
2. **Child Node:** The node which is the immediate successor of a node is called the child node of that node. Examples: {D, E} are the child nodes of {B}.
3. **Root Node:** The topmost node of a tree or the node which does not have any parent node is called the root node. {A} is the root node of the tree. A non-empty tree must contain exactly one root node and exactly one path from the root to all other nodes of the tree.
4. **Leaf Node or External Node:** The nodes which do not have any child nodes are called leaf nodes. {K, L, M, N, O, P} are the leaf nodes of the tree.
5. **Ancestor of a Node:** Any predecessor nodes on the path of the root to that node are called Ancestors of that node. {A, B} are the ancestor nodes of the node {E}
6. **Descendant:** Any successor node on the path from the leaf node to that node. {E, I} are the descendants of the node {B}.
7. **Sibling:** Children of the same parent node are called siblings. {D, E} are called siblings.
8. **Level of a node:** The count of edges on the path from the root node to that node. The root node has level 0.
9. **Internal node:** A node with at least one child is called Internal Node.
10. **Neighbour of a Node:** Parent or child nodes of that node are called neighbors of that node.
11. **Subtree:** Any node of the tree along with its descendant.

```
# Demonstration of Tree Basic Terminologies
# Function to add an edge between vertices x and y

# Function to print the parent of each node
```

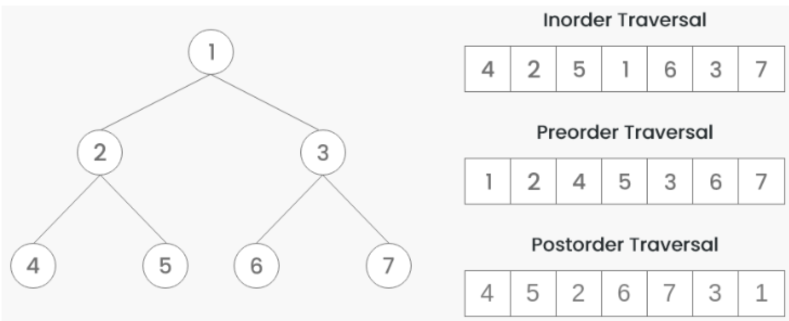
```
def printParents(node, adj, parent):
    # Write code here
    …

# Function to print the children of each node
def printChildren(Root, adj):
    # Write code here
    …

# Function to print the leaf nodes
def printLeafNodes(Root, adj):
    # Write code here
    …

# Function to print the degrees of each node
def printDegrees(Root, adj):
    # Write code here
    …

# Driver code
# Number of nodes
N = 7
Root = 1

# Adjacency list to store the tree
adj = []
for i in range(0, N+1):
    adj.append([])

# Creating the tree
adj[1].append(2)
adj[2].append(1)

adj[1].append(3)
adj[3].append(1)

adj[1].append(4)
adj[4].append(1)

adj[2].append(5)
adj[5].append(2)

adj[2].append(6)
adj[6].append(2)

adj[4].append(7)
adj[7].append(4)

# Printing the parents of each node
print("The parents of each node are:")
printParents(Root, adj, 0)
# Printing the children of each node
print("The children of each node are:")
printChildren(Root, adj)

# Printing the leaf nodes in the tree
```

```
print("The leaf nodes of the tree are:")
printLeafNodes(Root, adj)

# Printing the degrees of each node
print("The degrees of each node are:")
printDegrees(Root, adj)
```

9.2 Binary Tree Traversal Techniques

A binary tree data structure can be traversed in following ways:

1. Inorder Traversal
2. Preorder Traversal
3. Postorder Traversal
4. Level Order Traversal



Algorithm Inorder (tree)

1. Traverse the left subtree, i.e., call Inorder(left->subtree)
2. Visit the root.
3. Traverse the right subtree, i.e., call Inorder(right->subtree)

Algorithm Preorder (tree)

1. Visit the root.
2. Traverse the left subtree, i.e., call Preorder(left->subtree)
3. Traverse the right subtree, i.e., call Preorder(right->subtree)

Algorithm Postorder (tree)

1. Traverse the left subtree, i.e., call Postorder(left->subtree)
2. Traverse the right subtree, i.e., call Postorder(right->subtree)
3. Visit the root.

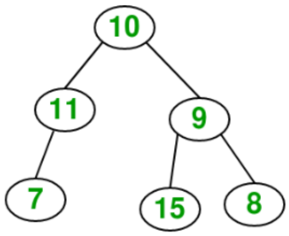
```
# Program to create a binary tree and print traversal orders
class Node:
    def __init__(self,data):
        self.data=data
        self.l=None
        self.r=None

class BT:
    def __init__(self):
        self.root=None
```

```
    def insert(self,n):
        # Write code here
        …

       def postorder(self,root):
        # Write code here
        …

    def preorder(self,root):
        # Write code here
        …

    def inorder(self,root):
        # Write code here
        …

# Driver code
b=BT()
print("*****************TREE USING DOUBLE LINKED LIST************************")
while True:
    print("1.Insert data to tree")
    print("2.Post Order Traversal")
    print("3.Pre Order Traversal")
    print("4.In Order Traversal")
    print("5.Exit")
    ch=int(input("Enter choice:"))
    if ch==1:
        n=int(input("Enter number of nodes:"))
        b.insert(n)
    elif ch==2:
        b.postorder(b.root)
    elif ch==3:
        b.preorder(b.root)
    elif ch==4:
        b.inorder(b.root)
    else:
        print("Exit")
        break
```

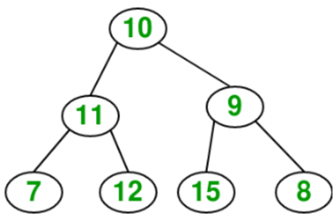
9.3 Insertion in a Binary Tree in Level Order

Given a binary tree and a key, insert the key into the binary tree at the first position available in level order.

Input: Consider the tree given below



Output:



After inserting 12

The idea is to do an iterative level order traversal of the given tree using queue. If we find a node whose left child is empty, we make a new key as the left child of the node. Else if we find a node whose right child is empty, we make the new key as the right child. We keep traversing the tree until we find a node whose either left or right child is empty.

```
# Insert element in binary tree
class newNode():
    def __init__(self, data):
        self.key = data
        self.left = None
        self.right = None

# Inorder traversal of a binary tree
def inorder(temp):
    # Write code here
    …

# function to insert element in binary tree
def insert(temp,key):
   # Write code here
   …

# Driver code
root = newNode(10)
root.left = newNode(11)
root.left.left = newNode(7)
root.right = newNode(9)
root.right.left = newNode(15)
root.right.right = newNode(8)
print("Inorder traversal before insertion:", end = " ")
inorder(root)

key = 12
insert(root, key)
```

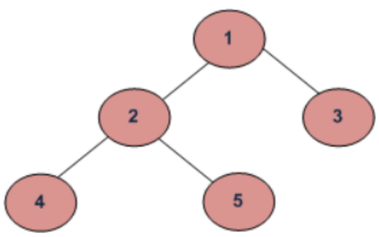
```
print()
print("Inorder traversal after insertion:", end = " ")
inorder(root)
```

9.4 Finding the Maximum Height or Depth of a Binary Tree

Given a binary tree, the task is to find the height of the tree. The height of the tree is the number of edges in the tree from the root to the deepest node.

Note: The height of an empty tree is 0.

Input: Consider the tree below



Recursively calculate the height of the left and the right subtrees of a node and assign height to the node as max of the heights of two children plus 1.

maxDepth('1') = max(maxDepth('2'), maxDepth('3')) + 1 = 2 + 1
because recursively
maxDepth('2') = max (maxDepth('4'), maxDepth('5')) + 1 = 1 + 1 and (as height of both '4' and '5' are 1)
maxDepth('3') = 1

Procedure:

- Recursively do a Depth-first search.
- If the tree is empty then return 0
- Otherwise, do the following
  - Get the max depth of the left subtree recursively i.e. call maxDepth(tree->left-subtree)
  - Get the max depth of the right subtree recursively i.e. call maxDepth(tree->right-subtree)
  - Get the max of max depths of left and right subtrees and add 1 to it for the current node.

$$max_depth = max(maxdeptofleftsubtree, maxdepthofrightsubtree) + 1$$

  - Return max_depth.

```
# Find the maximum depth of tree
# A binary tree node
class Node:
    # Constructor to create a new node
    def __init__(self, data):
        self.data = data
        self.left = None
```

```
        self.right = None

# Compute the "maxDepth" of a tree -- the number of nodes
# along the longest path from the root node down to the farthest leaf node

def maxDepth(node):
    # Write code here
    …

# Driver program to test above function
root = Node(1)
root.left = Node(2)
root.right = Node(3)
root.left.left = Node(4)
root.left.right = Node(5)
print("Height of tree is %d" % (maxDepth(root)))
```

9.5 Deletion in a Binary Tree

Given a binary tree, delete a node from it by making sure that the tree shrinks from the bottom (i.e. the deleted node is replaced by the bottom-most and rightmost node).

Input: Delete 10 in below tree

```
     10
    /  \
  20    30
```

Output:

```
     30
    /
  20
```

Input: Delete 20 in below tree

```
     10
    /  \
  20    30
          \
           40
```

Output:

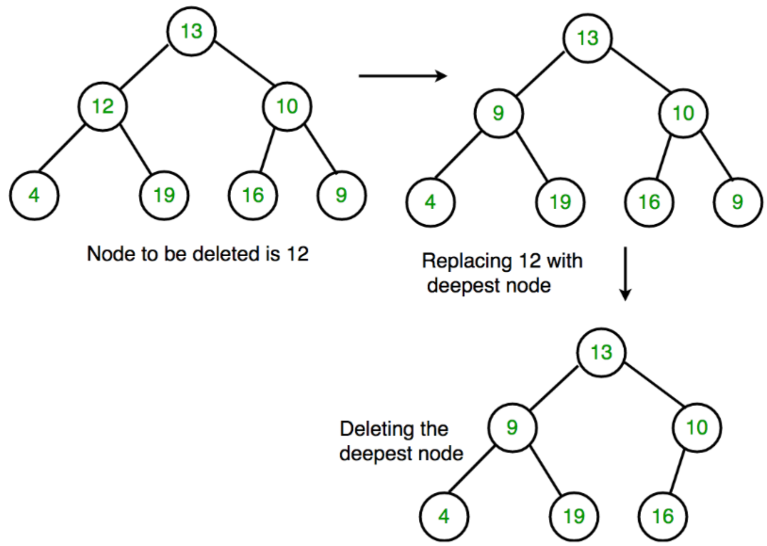
```
     10
   /    \
 40      30
```

Algorithm:

1. Starting at the root, find the deepest and rightmost node in the binary tree and the node which we want to delete.

2. Replace the deepest rightmost node's data with the node to be deleted.

3. Then delete the deepest rightmost node.




```
# Deletion in a Binary Tree

# Create a node with data, left child and right child.

class Node:
    def __init__(self, data):
        self.data = data
        self.left = None
        self.right = None

# Inorder traversal of a binary tree
def inorder(temp):
    # Write code here
    …

# function to delete the given deepest node (d_node) in binary tree
def deleteDeepest(root, d_node):
    # Write code here
    …

# function to delete element in binary tree
def deletion(root, key):
    # Write code here
    …

# Driver code
root = Node(10)
root.left = Node(11)
root.left.left = Node(7)
root.left.right = Node(12)
root.right = Node(9)
root.right.left = Node(15)
root.right.right = Node(8)
print("The tree before the deletion: ", end = "")
inorder(root)
key = 11
```

```
root = deletion(root, key)
print();
print("The tree after the deletion: ", end = "")
inorder(root)
```

10. Binary Search Tree (BST)

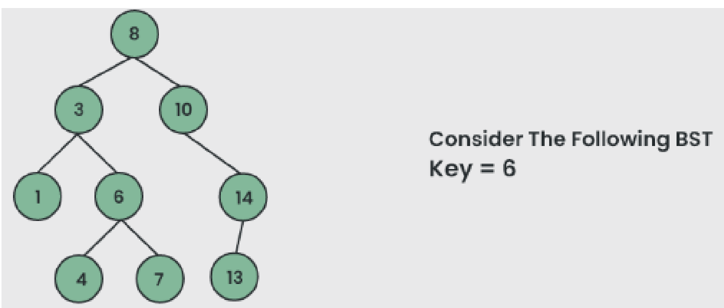
10.1 Searching in Binary Search Tree

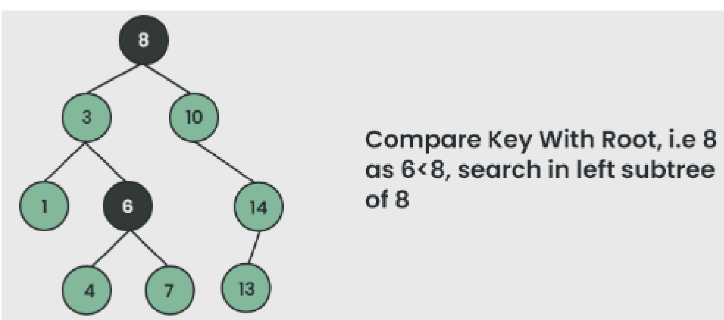
Given a BST, the task is to delete a node in this BST. For searching a value in BST, consider it as a sorted array. Perform search operation in BST using Binary Search Algorithm.

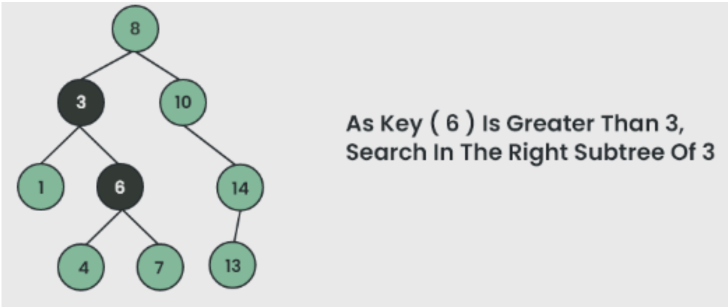
Algorithm to search for a key in a given Binary Search Tree:

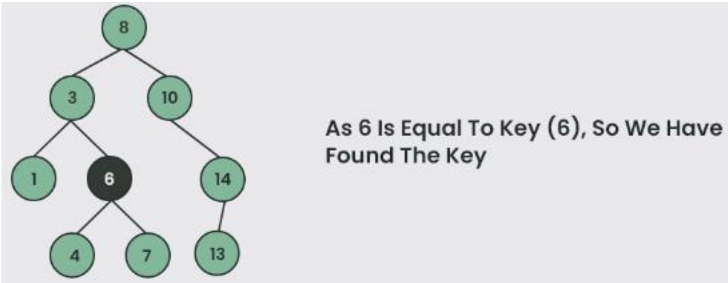
Let's say we want to search for the number **X,** We start at the root. Then:

- We compare the value to be searched with the value of the root.
- If it's equal we are done with the search if it's smaller we know that we need to go to the left subtree because in a binary search tree all the elements in the left subtree are smaller and all the elements in the right subtree are larger.
- Repeat the above step till no more traversal is possible
- If at any iteration, key is found, return True. Else False.









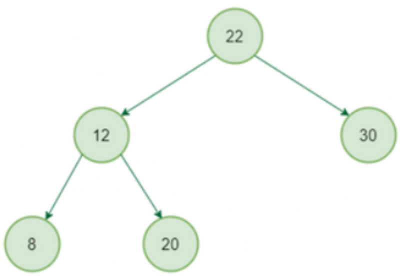

```
# Search a given key in a given BST
 class Node:
    # Constructor to create a new node
    def __init__(self, key):
        self.key = key
        self.left = None
        self.right = None

# A utility function to insert
# a new node with the given key in BST
def insert(node, key):
    # Write code here
    …
# Utility function to search a key in a BST
def search(root, key):
    # Write code here
    …
# Driver Code
root = None
root = insert(root, 50)
insert(root, 30)
insert(root, 20)
insert(root, 40)
insert(root, 70)
insert(root, 60)
insert(root, 80)
# Key to be found
key = 6

# Searching in a BST
if search(root, key) is None:
    print(key, "not found")

else:
```

```
    print(key, "found")

key = 60

# Searching in a BST
if search(root, key) is None:
    print(key, "not found")
else:
    print(key, "found")
```

10.2 Find the node with Minimum Value in a BST

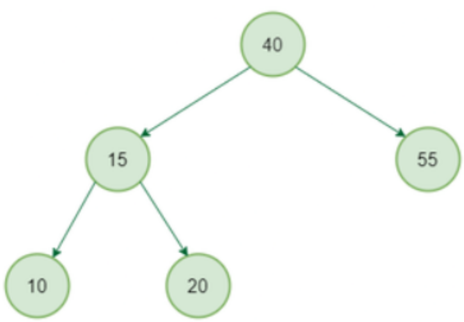
Write a function to find the node with minimum value in a Binary Search Tree.

Input: Consider the tree given below



Output: 8

Input: Consider the tree given below



Output: 10

```
from typing import List
class Node:
    def __init__(self, data):
        self.data = data
        self.left = None
        self.right = None

# Give a binary search tree and a number, inserts a new node with the given number
# in the correct place in the tree. Returns the new root pointer
def insert(node: Node, data: int) -> Node:
    # Write code here
    …
```

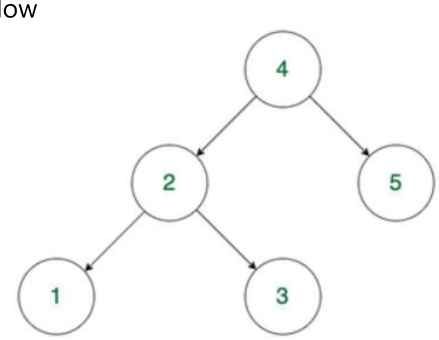
```
# Given a non-empty binary search tree, inorder traversal for
# the tree is stored in the list sorted_inorder. Inorder is LEFT, ROOT, RIGHT.

def inorder(node: Node, sorted_inorder: List[int]) -> None:
    # Write code here
    …
# Driver Code
root = None
root = insert(root, 4)
insert(root, 2)
insert(root, 1)
insert(root, 3)
insert(root, 6)
insert(root, 4)
insert(root, 5)
sorted_inorder = []
inorder(root, sorted_inorder)  # calling the recursive function

# Values of all nodes will appear in sorted order in the list sorted_inorder
print(f"Minimum value in BST is {sorted_inorder[0]}")
```

10.3 Check if a Binary Tree is BST or not

A binary search tree (BST) is a node-based binary tree data structure that has the following properties.

1. The left subtree of a node contains only nodes with keys less than the node's key.
2. The right subtree of a node contains only nodes with keys greater than the node's key.
3. Both the left and right subtrees must also be binary search trees.
4. Each node (item in the tree) has a distinct key.

Input: Consider the tree given below



Output: Check if max value in left subtree is smaller than the node and min value in right subtree greater than the node, then print it "Is BST" otherwise "Not a BST"

Procedure:

1. If the current node is null then return true
2. If the value of the left child of the node is greater than or equal to the current node then return false
3. If the value of the right child of the node is less than or equal to the current node then return false
4. If the left subtree or the right subtree is not a BST then return false
5. Else return true

```
# Program to check if a binary tree is BST or not
# A binary tree node has data, pointer to left child and a pointer to right child

class Node:
     def __init__(self, data):
        self.data = data
        self.left = None
        self.right = None

def maxValue(node):
    # Write code here
    …
   def minValue(node):
    # Write code here
    …
 # Returns true if a binary tree is a binary search tree
def isBST(node):
    # Write code here
    …
 # Driver code
root = Node(4)
root.left = Node(2)
root.right = Node(5)
# root.right.left = Node(7)
root.left.left = Node(1)
root.left.right = Node(3)

# Function call
if isBST(root) is True:
    print("Is BST")
else:
    print("Not a BST")
```

10.4 Second Largest Element in BST

Given a Binary search tree (BST), find the second largest element.

Input: Root of below BST

```
          10
         /
        5
```

Output: 5

Input: Root of below BST

```
         10

        /  \

       5    20

              \

               30
```

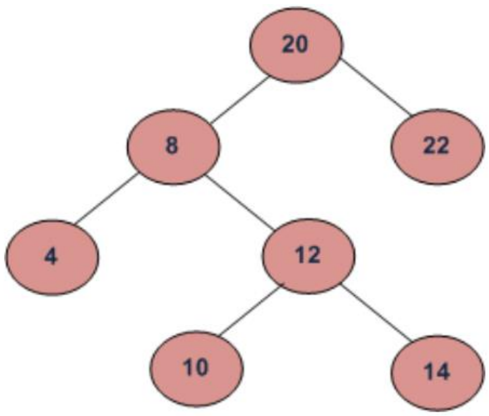
Output: 20

Procedure: The second largest element is second last element in inorder traversal and second element in reverse inorder traversal. We traverse given Binary Search Tree in reverse inorder and keep track of counts of nodes visited. Once the count becomes 2, we print the node.

```
# Find the second largest element in
class Node:

    # Constructor to create a new node
    def __init__(self, data):
        self.key = data
        self.left = None
        self.right = None

# A function to find 2nd largest element in a given tree.
def secondLargestUtil(root, c):
    # Write code here
    …
# Function to find 2nd largest element
def secondLargest(root):
    # Write code here
    …

# A utility function to insert a new node with given key in BST
def insert(node, key):

# Driver Code
# Let us create following BST
#       50
#     /    \
#    30     70
#   / \     / \
#  20  40  60  80

root = None
root = insert(root, 50)
insert(root, 30)
insert(root, 20)
insert(root, 40)
insert(root, 70)
insert(root, 60)
insert(root, 80)
secondLargest(root)
```

Try:

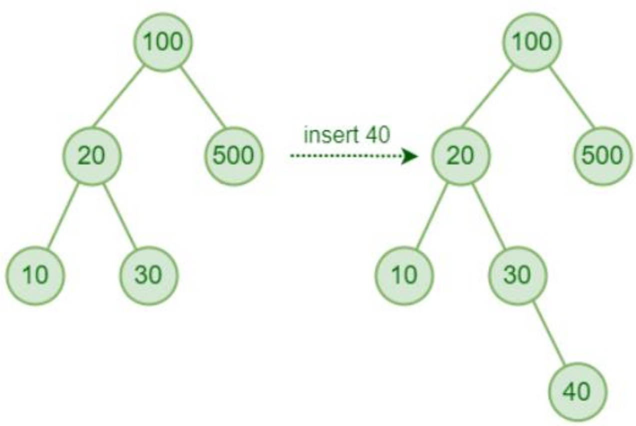
1. **Kth largest element in BST when modification to BST is not allowed:** Given a Binary Search Tree (BST) and a positive integer k, find the k'th largest element in the Binary Search Tree. For a given BST, if k = 3, then output should be 14, and if k = 5, then output should be 10.



10.5 Insertion in Binary Search Tree (BST)

Given a Binary search tree (BST), the task is to insert a new node in this BST.

Input: Consider a BST and insert the element 40 into it.



Procedure for inserting a value in a BST:

A new key is always inserted at the leaf by maintaining the property of the binary search tree. We start searching for a key from the root until we hit a leaf node. Once a leaf node is found, the new node is added as a child of the leaf node. The below steps are followed while we try to insert a node into a binary search tree:

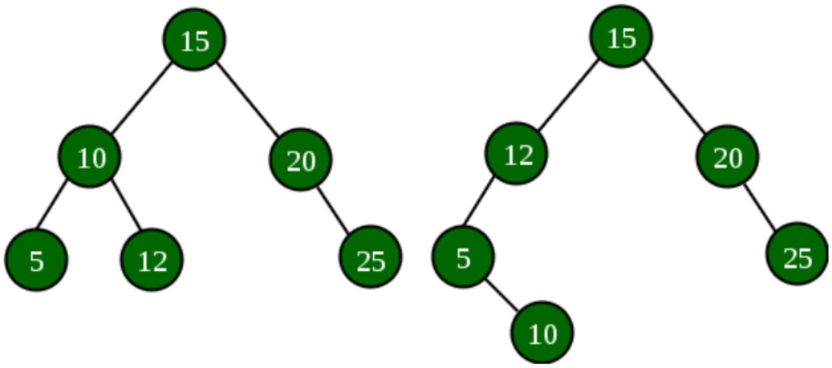
- Check the value to be inserted (say X) with the value of the current node (say val) we are in:
  - If X is less than val move to the left subtree.
  - Otherwise, move to the right subtree.
  - Once the leaf node is reached, insert X to its right or left based on the relation between X and the leaf node's value.

```python
# insert operation in binary search tree
# A utility class that represents an individual node in a BST
class Node:
    def __init__(self, key):
        self.left = None
        self.right = None
        self.val = key

 # A utility function to insert a new node with the given key
def insert(root, key):
    # Write code here
    …

 # A utility function to do inorder tree traversal
def inorder(root):
    # Write code here
    …

# Driver code
# Let us create the following BST
#       50
#     /     \
#    30     70
#   / \     / \
#  20 40   60 80
r = Node(50)
r = insert(r, 30)
r = insert(r, 20)
r = insert(r, 40)
r = insert(r, 70)
r = insert(r, 60)
r = insert(r, 80)

# Print inorder traversal of the BST
inorder(r)
```

Try:

1. **Check if two BSTs contain same set of elements:** Given two Binary Search Trees consisting of unique positive elements, we have to check whether the two BSTs contain the same set of elements or not.

Input: Consider two BSTs which contains same set of elements {5, 10, 12, 15, 20, 25}, but the structure of the two given BSTs can be different.



11. AVL Tree

11.1 Insertion in an AVL Tree

AVL tree is a self-balancing Binary Search Tree (BST) where the difference between heights of left and right subtrees cannot be more than one for all nodes. To make sure that the given tree remains AVL after every insertion, we must augment the standard BST insert operation to perform some re-balancing. Following are two basic operations that can be performed to balance a BST without violating the BST property (keys(left) < key(root) < keys(right)).

- Left Rotation
- Right Rotation

T1, T2 and T3 are subtrees of the tree, rooted with y (on the left side) or x (on the right side)

```
     y                               x
    / \     Right Rotation          /  \
   x   T3   - - - - - - - >        T1   y
  / \       < - - - - - - -            / \
 T1  T2     Left Rotation            T2  T3
```

Keys in both of the above trees follow the following order

keys(T1) < key(x) < keys(T2) < key(y) < keys(T3)

So BST property is not violated anywhere.

Procedure for inserting a node into an AVL tree

Let the newly inserted node be w

- Perform standard BST insert for w.
- Starting from w, travel up and find the first unbalanced node. Let z be the first unbalanced node, y be the child of z that comes on the path from w to z and x be the grandchild of z that comes on the path from w to z.

- Re-balance the tree by performing appropriate rotations on the subtree rooted with z. There can be 4 possible cases that need to be handled as x, y and z can be arranged in 4 ways.
- Following are the possible 4 arrangements:
  - y is the left child of z and x is the left child of y (Left Left Case)
  - y is the left child of z and x is the right child of y (Left Right Case)
  - y is the right child of z and x is the right child of y (Right Right Case)
  - y is the right child of z and x is the left child of y (Right Left Case)

```
# Insert a node in AVL tree

# Generic tree node class
class TreeNode(object):
    def __init__(self, val):
        self.val = val
        self.left = None
        self.right = None
        self.height = 1

# AVL tree class which supports the insert operation
class AVL_Tree(object):

    # Recursive function to insert key in subtree rooted with node and returns
    # new root of subtree.
    def insert(self, root, key):
        # Write code here
        …

    def leftRotate(self, z):
        # Write code here
        …

    def rightRotate(self, z):
        # Write code here
        …

    def getHeight(self, root):
        # Write code here
        …

    def getBalance(self, root):
        # Write code here
        …

    def preOrder(self, root):
        # Write code here
        …

# Driver code
myTree = AVL_Tree()
```

```
root = None

root = myTree.insert(root, 10)
root = myTree.insert(root, 20)
root = myTree.insert(root, 30)
root = myTree.insert(root, 40)
root = myTree.insert(root, 50)
root = myTree.insert(root, 25)

"""The constructed AVL Tree would be
            30
           /  \
         20   40
        /  \     \
       10  25    50"""

# Preorder Traversal
print("Preorder traversal of the",
      "constructed AVL tree is")
myTree.preOrder(root)
print()
```

11.2 Deletion in an AVL Tree

Given an AVL tree, make sure that the given tree remains AVL after every deletion, we must augment the standard BST delete operation to perform some re-balancing. Following are two basic operations that can be performed to re-balance a BST without violating the BST property (keys(left) < key(root) < keys(right)).

1. Left Rotation
2. Right Rotation

T1, T2 and T3 are subtrees of the tree rooted with y (on left side)
or x (on right side)

```
     y                               x
    / \     Right Rotation          /  \
   x   T3   - - - - - - - >        T1   y
  / \       < - - - - - - -            / \
T1   T2     Left Rotation            T2   T3
```

Keys in both of the above trees follow the following order
   keys(T1) < key(x) < keys(T2) < key(y) < keys(T3)
So BST property is not violated anywhere.

Procedure to delete a node from AVL tree:

Let w be the node to be deleted
1. Perform standard BST delete for w.

2. Starting from w, travel up and find the first unbalanced node. Let z be the first unbalanced node, y be the larger height child of z, and x be the larger height child of y. Note that the definitions of x and y are different from insertion here.

3. Re-balance the tree by performing appropriate rotations on the subtree rooted with z. There can be 4 possible cases that needs to be handled as x, y and z can be arranged in 4 ways. Following are the possible 4 arrangements:

   i. y is left child of z and x is left child of y (Left Left Case)

   ii. y is left child of z and x is right child of y (Left Right Case)

   iii. y is right child of z and x is right child of y (Right Right Case)

   iv. y is right child of z and x is left child of y (Right Left Case)

```
# delete a node in AVL tree

class TreeNode(object):
    def __init__(self, val):
        self.val = val
        self.left = None
        self.right = None
        self.height = 1

# AVL tree class which supports insertion, deletion operations
class AVL_Tree(object):

    def insert(self, root, key):
        # Write code here
        …

    # Recursive function to delete a node with given key from subtree
    # with given root. It returns root of the modified subtree.
    def delete(self, root, key):
        # Write code here
        …

    def leftRotate(self, z):
        # Write code here
        …

    def rightRotate(self, z):
        # Write code here
        …

    def getHeight(self, root):
        # Write code here
        …

    def getBalance(self, root):
        # Write code here
        …
```

```
    def getMinValueNode(self, root):
        # Write code here
        …

    def preOrder(self, root):
        # Write code here
        …

myTree = AVL_Tree()
root = None
nums = [9, 5, 10, 0, 6, 11, -1, 1, 2]

for num in nums:
    root = myTree.insert(root, num)


# Preorder Traversal
print("Preorder Traversal after insertion -")
myTree.preOrder(root)
print()

# Delete
key = 10
root = myTree.delete(root, key)

# Preorder Traversal
print("Preorder Traversal after deletion -")
myTree.preOrder(root)
print()
```

11.3 Count Greater Nodes in AVL Tree

Given an AVL tree, calculate number of elements which are greater than given value in AVL tree.

Input: x = 5

Root of below AVL tree

```
        9
       / \
      1   10
     / \    \
    0   5    11
   /   / \
 -1   2   6
```

Output: 4

Explanation: There are 4 values which are greater than 5 in AVL tree which are 6, 9, 10 and 11.

```
# Count greater nodes in an AVL tree

class Node:
    def __init__(self, key):
        self.key = key
        self.left = None
        self.right = None
        self.height = 1
        self.desc = 0

 def height(N):
    if N is None:
        return 0
    return N.height

# A utility function to get maximum of two integers

def max(a, b):
    if a > b:
        return a
    return b

def newNode(key):
    # Write code here
    …

# A utility function to right rotate subtree rooted with y

def rightRotate(y):
    # Write code here
    …

def leftRotate(x):
    # Write code here
    …
def getBalance(N):
    # Write code here
    …

def insert(root, key):
    # Write code here
    …

def minValueNode(node):
    # Write code here
    …

# Recursive function to delete a node with given key # from subtree with given root. It
returns root of the modified subtree.

def deleteNode(root, key):
    # Write code here
    …

def preOrder(root):
```

```
    # Write code here
    …

def CountGreater(root, x):
    # Write code here
    …

# Driver program to test above function
root = None
root = insert(root, 9)
root = insert(root, 5)
root = insert(root, 10)
root = insert(root, 0)
root = insert(root, 6)
root = insert(root, 11)
root = insert(root, -1)
root = insert(root, 1)
root = insert(root, 2)

print("Preorder traversal of the constructed AVL tree is")
preOrder(root)

print("Number of elements greater than 9 are")
print(CountGreater(root, 9))

root = deleteNode(root, 10)

print("Preorder traversal after deletion of 10")
preOrder(root)
print('Number of elements greater than 9 are')
print(CountGreater(root, 9))
```

11.4 Minimum Number of Nodes in an AVL Tree with given Height

Given the height of an AVL tree 'h', the task is to find the minimum number of nodes the tree can have.

Input: H = 0

Output: N = 1

Only '1' node is possible if the height
of the tree is '0' which is the root node.

Input: H = 3

Output: N = 7

Recursive approach:
In an AVL tree, we have to maintain the height balance property, i.e. difference in the height of the left and the right subtrees cannot be other than -1, 0 or 1 for each node.

We will try to create a recurrence relation to find minimum number of nodes for a given height, n(h).

- For height = 0, we can only have a single node in an AVL tree, i.e. n(0) = 1
- For height = 1, we can have a minimum of two nodes in an AVL tree, i.e. n(1) = 2

- Now for any height ‘h’, root will have two subtrees (left and right). Out of which one has to be of height h-1 and other of h-2. [root node excluded]

- So, n(h) = 1 + n(h-1) + n(h-2) is the required recurrence relation for h>=2 [1 is added for the root node]

```
# Function to find minimum number of nodes

def AVLnodes(height):
    # Write code here
    …
# Driver Code
H = 3
print(AVLnodes(H))
```

12. Graph Traversal

12.1 Breadth First Search

The **Breadth First Search (BFS)** algorithm is used to search a graph data structure for a node that meets a set of criteria. It starts at the root of the graph and visits all nodes at the current depth level before moving on to the nodes at the next depth level.
For a given graph G, print BFS traversal from a given source vertex.

```
# BFS traversal from a given source vertex.

from collections import defaultdict
 # This class represents a directed graph using adjacency list representation
class Graph:

    # Constructor
    def __init__(self):
        # Default dictionary to store graph
        self.graph = defaultdict(list)

    # Function to add an edge to graph
    def addEdge(self, u, v):
        self.graph[u].append(v)

    # Function to print a BFS of graph
    def BFS(self, s):
      # Write code here
      …

# Create a graph given in the above diagram
g = Graph()
g.addEdge(0, 1)
g.addEdge(0, 2)
g.addEdge(1, 2)
g.addEdge(2, 0)
g.addEdge(2, 3)
g.addEdge(3, 3)

print("Following is Breadth First Traversal" " (starting from vertex 2)")
```

```
g.BFS(2)
```

Output: Following is Breadth First Traversal (starting from vertex 2)
2 0 3 1

12.2 Depth First Search

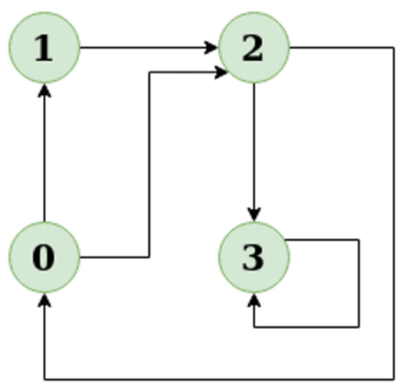
Depth First Traversal (or DFS) for a graph is similar to Depth First Traversal of a tree. The only catch here is, that, unlike trees, graphs may contain cycles (a node may be visited twice). To avoid processing a node more than once, use a boolean visited array. A graph can have more than one DFS traversal.

For a given graph G, print DFS traversal from a given source vertex.

Input: n = 4, e = 6
0 -> 1, 0 -> 2, 1 -> 2, 2 -> 0, 2 -> 3, 3 -> 3

Output: DFS from vertex 1: 1 2 0 3

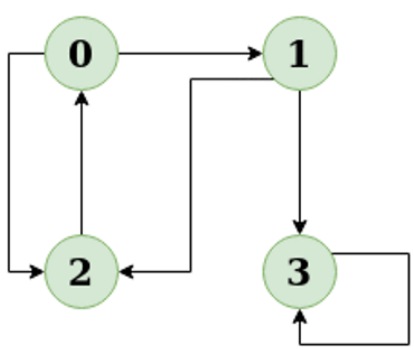
Explanation:
DFS Diagram:



Input: n = 4, e = 6
2 -> 0, 0 -> 2, 1 -> 2, 0 -> 1, 3 -> 3, 1 -> 3

Output: DFS from vertex 2: 2 0 1 3

Explanation:
DFS Diagram:



```
# DFS traversal from a given graph
from collections import defaultdict

# This class represents a directed graph using adjacency list representation
class Graph:
    # Constructor
    def __init__(self):
        # Default dictionary to store graph
        self.graph = defaultdict(list)

    # Function to add an edge to graph
    def addEdge(self, u, v):
        self.graph[u].append(v)
    # A function used by DFS
    def DFSUtil(self, v, visited):
        # Write code here
        …

    # The function to do DFS traversal. It uses recursive DFSUtil()

    def DFS(self, v):
       # Write code here
       …
# Driver's code
g = Graph()
g.addEdge(0, 1)
g.addEdge(0, 2)
g.addEdge(1, 2)
g.addEdge(2, 0)
g.addEdge(2, 3)
g.addEdge(3, 3)
print("Following is Depth First Traversal (starting from vertex 2)")
# Function call
g.DFS(2)
```

12.3 Best First Search (Informed Search)

The idea of Best First Search is to use an evaluation function to decide which adjacent is most promising and then explore. Best First Search falls under the category of Heuristic Search or Informed Search.

Implementation of Best First Search:

We use a priority queue or heap to store the costs of nodes that have the lowest evaluation function value. So the implementation is a variation of BFS, we just need to change Queue to PriorityQueue.

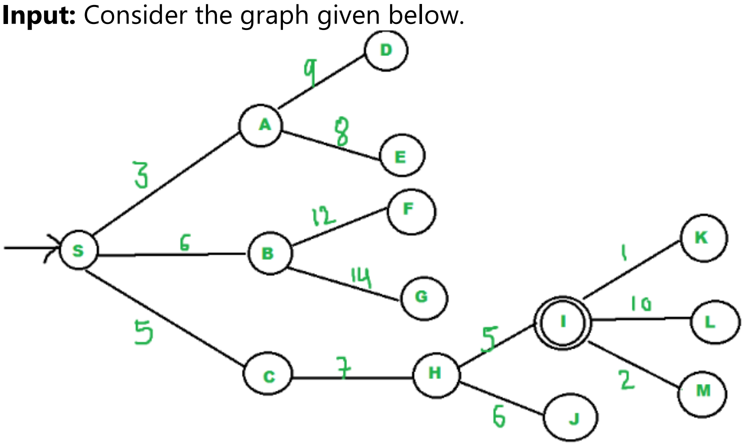
Algorithm:

Best-First-Search(Graph g, Node start)

1) Create an empty PriorityQueue
   PriorityQueue pq;
2) Insert "start" in pq.
   pq.insert(start)
3) Until PriorityQueue is empty
     u = PriorityQueue.DeleteMin
     If u is the goal
        Exit

```
        Else
            Foreach neighbor v of u
                If v "Unvisited"
                    Mark v "Visited"
                    pq.insert(v)
            Mark u "Examined"
End procedure
```

Input: Consider the graph given below.



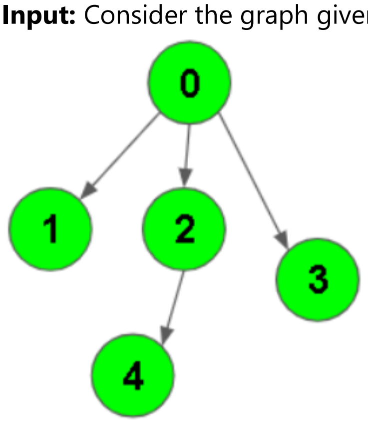
- We start from source "S" and search for goal "I" using given costs and Best First search.
- pq initially contains S
    - We remove S from pq and process unvisited neighbors of S to pq.
    - pq now contains {A, C, B} (C is put before B because C has lesser cost)
- We remove A from pq and process unvisited neighbors of A to pq.
    - pq now contains {C, B, E, D}
- We remove C from pq and process unvisited neighbors of C to pq.
    - pq now contains {B, H, E, D}
- We remove B from pq and process unvisited neighbors of B to pq.
    - pq now contains {H, E, D, F, G}
- We remove H from pq.
- Since our goal "I" is a neighbor of H, we return.

```
from queue import PriorityQueue
v = 14
graph = [[] for i in range(v)]

# Function For Implementing Best First Search
# Gives output path having lowest cost

def best_first_search(actual_Src, target, n):
    # Write code here
    …
# Function for adding edges to graph
def addedge(x, y, cost):
    # Write code here
    …
# The nodes shown in above example(by alphabets) are
# implemented using integers addedge(x,y,cost);
addedge(0, 1, 3)
addedge(0, 2, 6)
addedge(0, 3, 5)
addedge(1, 4, 9)
addedge(1, 5, 8)
addedge(2, 6, 12)
addedge(2, 7, 14)
addedge(3, 8, 7)
addedge(8, 9, 5)
addedge(8, 10, 6)
addedge(9, 11, 1)
addedge(9, 12, 10)
addedge(9, 13, 2)

source = 0
target = 9
best_first_search(source, target, v)
```

12.4 Breadth First Traversal of a Graph

Given a directed graph. The task is to do Breadth First Traversal of this graph starting from 0.
One can move from node u to node v only if there's an edge from u to v. Find the BFS traversal of the graph starting from the 0th vertex, from left to right according to the input graph. Also, you should only take nodes directly or indirectly connected from Node 0 in consideration.

Input: Consider the graph given below where V = 5, E = 4, edges = {(0,1), (0,2), (0,3), (2,4)}



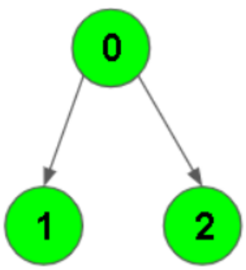
Output: 0 1 2 3 4

Explanation:

0 is connected to 1, 2, and 3.

2 is connected to 4.
So starting from 0, it will go to 1 then 2 then 3. After this 2 to 4, thus BFS will be 0 1 2 3 4.

Input: Consider the graph given below where V = 3, E = 2, edges = {(0, 1), (0, 2)}



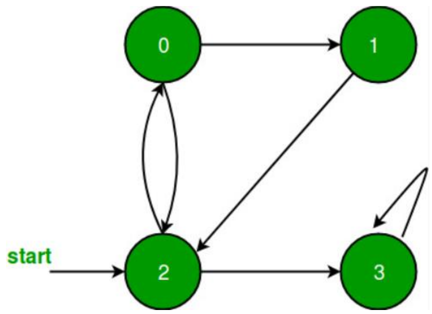
Output: 0 1 2
Explanation:
0 is connected to 1, 2. So starting from 0, it will go to 1 then 2, thus BFS will be 0 1 2.
Your task is to complete the function **bfsOfGraph()** which takes the integer V denoting the number of vertices and adjacency list as input parameters and returns a list containing the BFS traversal of the graph starting from the 0th vertex from left to right.

```
from typing import List
from queue import Queue
class Solution:

    # Function to return Breadth First Traversal of given graph.
    def bfsOfGraph(self, V: int, adj: List[List[int]]) -> List[int]:
        # Write code here
        …
# Driver Code
T=int(input())
for i in range(T):
      V, E = map(int, input().split())
      adj = [[] for i in range(V)]
      for _ in range(E):
            u, v = map(int, input().split())
            adj[u].append(v)
      ob = Solution()
      ans = ob.bfsOfGraph(V, adj)
      for i in range(len(ans)):
               print(ans[i], end = " ")
      print()
```

12.5 Depth First Search (DFS) for Disconnected Graph

Given a Disconnected Graph, the task is to implement DFS or Depth First Search Algorithm for this Disconnected Graph.

Input: Consider the graph given below.



Output: 0 1 2 3

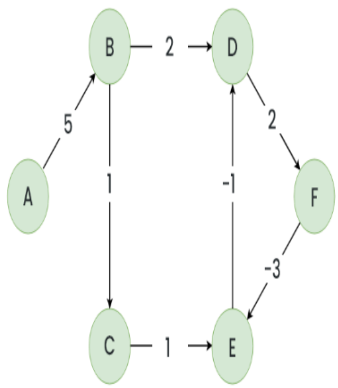
Procedure for DFS on Disconnected Graph:
Iterate over all the vertices of the graph and for any unvisited vertex, run a DFS from that vertex.

```
# DFS traversal for complete graph
from collections import defaultdict

# This class represents a directed graph using adjacency list representation
class Graph:
    # Constructor
    def __init__(self):
        # Default dictionary to store graph
        self.graph = defaultdict(list)

    # Function to add an edge to graph
    def addEdge(self, u, v):
        # Write code here
        …

    # A function used by DFS
    def DFSUtil(self, v, visited):
        # Write code here
        …

    # The function to do DFS traversal.
    # It uses recursive DFSUtil
    def DFS(self):
        # Write code here
        …
# Driver's code
print("Following is Depth First Traversal")
g = Graph()
g.addEdge(0, 1)
g.addEdge(0, 2)
g.addEdge(1, 2)
g.addEdge(2, 0)
g.addEdge(2, 3)
g.addEdge(3, 3)

# Function call
g.DFS()
```

Try:

1. **Detect a negative cycle in a Graph (Bellman Ford):** A Bellman-Ford algorithm is also guaranteed to find the shortest path in a graph, similar to Dijkstra's algorithm. Although Bellman-Ford is slower than Dijkstra's algorithm, it is capable of handling graphs with negative edge weights, which makes it more versatile. The shortest path cannot be found if there exists a negative cycle in the graph. If we continue to go around the negative cycle an infinite number of times, then the cost of the path will continue to decrease (even though the length of the path is increasing).

Consider a graph G and detect a negative cycle in the graph using Bellman Ford algorithm.



13. Minimum Spanning Tree (MST)

13.1 Kruskal's Algorithm

In Kruskal's algorithm, sort all edges of the given graph in increasing order. Then it keeps on adding new edges and nodes in the MST if the newly added edge does not form a cycle. It picks the minimum weighted edge at first and the maximum weighted edge at last.

MST using Kruskal's algorithm:

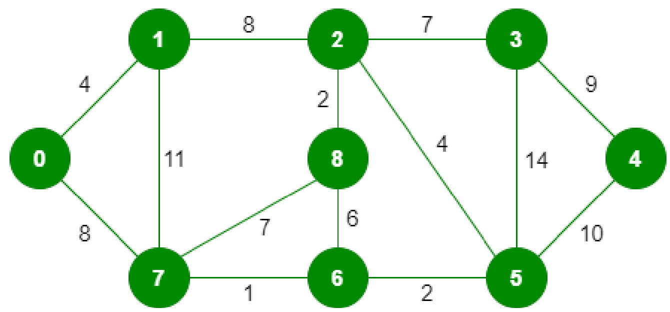
1. Sort all the edges in non-decreasing order of their weight.

2. Pick the smallest edge. Check if it forms a cycle with the spanning tree formed so far. If the cycle is not formed, include this edge. Else, discard it.

3. Repeat step#2 until there are (V-1) edges in the spanning tree.

Kruskal's algorithm to find the minimum cost spanning tree uses the greedy approach. The Greedy Choice is to pick the smallest weight edge that does not cause a cycle in the MST constructed so far.

Input: For the given graph G find the minimum cost spanning tree.



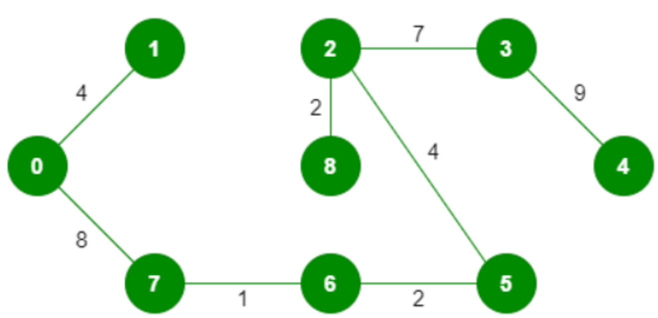
The graph contains 9 vertices and 14 edges. So, the minimum spanning tree formed will be having (9 – 1) = 8 edges.

After sorting:

Weight	Source	Destination
1	7	6
2	8	2
2	6	5
4	0	1
4	2	5
6	8	6
7	2	3
7	7	8
8	0	7
8	1	2
9	3	4
10	5	4
11	1	7
14	3	5

Now pick all edges one by one from the sorted list of edges.

Output:




```
# Kruskal's algorithm to find minimum Spanning Tree of a given connected,
# undirected and weighted graph

# Class to represent a graph
```

```
class Graph:
    def __init__(self, vertices):
        self.V = vertices
        self.graph = []

    # Function to add an edge to graph
    def addEdge(self, u, v, w):
        self.graph.append([u, v, w])

    def find(self, parent, i):
        …

    def union(self, parent, rank, x, y):
        …

    def KruskalMST(self):
      # write your code here
      …

# Driver code
g = Graph(4)
g.addEdge(0, 1, 10)
g.addEdge(0, 2, 6)
g.addEdge(0, 3, 5)
g.addEdge(1, 3, 15)
g.addEdge(2, 3, 4)

# Function call
g.KruskalMST()
```

Output: Following are the edges in the constructed MST
2 -- 3 == 4

0 -- 3 == 5

0 -- 1 == 10

Minimum Cost Spanning Tree: 19

13.2 Prim's Algorithm

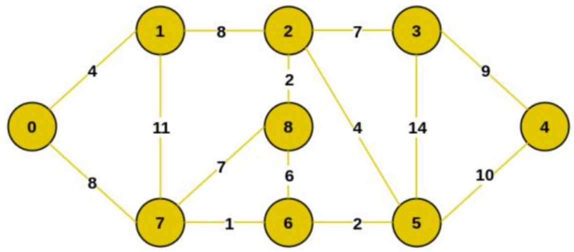
The Prim's algorithm starts with an empty spanning tree. The idea is to maintain two sets of vertices. The first set contains the vertices already included in the MST, and the other set contains the vertices not yet included. At every step, it considers all the edges that connect the two sets and picks the minimum weight edge from these edges. After picking the edge, it moves the other endpoint of the edge to the set containing MST.

Prim's Algorithm:
The working of Prim's algorithm can be described by using the following steps:

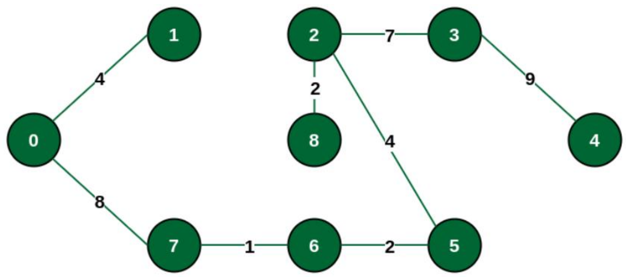
1. Determine an arbitrary vertex as the starting vertex of the MST.
2. Follow steps 3 to 5 till there are vertices that are not included in the MST (known as fringe vertex).
3. Find edges connecting any tree vertex with the fringe vertices.
4. Find the minimum among these edges.

5. Add the chosen edge to the MST if it does not form any cycle.
6. Return the MST and exit

Input: For the given graph G find the minimum cost spanning tree.



Output: The final structure of the MST is as follows and the weight of the edges of the MST is (4 + 8 + 1 + 2 + 4 + 2 + 7 + 9) = 37.




```
# Prim's Minimum Spanning Tree (MST) algorithm.
# The program is for adjacency matrix representation of the graph

# Library for INT_MAX
import sys

class Graph():
    def __init__(self, vertices):
        self.V = vertices
        self.graph = [[0 for column in range(vertices)]
                      for row in range(vertices)]

    # A utility function to print
    # the constructed MST stored in parent[]
    def printMST(self, parent):
        print("Edge \tWeight")
        for i in range(1, self.V):
            print(parent[i], "-", i, "\t", self.graph[i][parent[i]])

    # A utility function to find the vertex with
    # minimum distance value, from the set of vertices
    # not yet included in shortest path tree
    def minKey(self, key, mstSet):
       # write your code here
       …

    def primMST(self):
       # write your code here
       …
```

```
# Driver's code
g = Graph(5)
g.graph = [[0, 2, 0, 6, 0],
           [2, 0, 3, 8, 5],
           [0, 3, 0, 0, 7],
           [6, 8, 0, 0, 9],
           [0, 5, 7, 9, 0]]

g.primMST()
```

Output:

Edge Weight

0 - 1 2

1 - 2 3

0 - 3 6

1 - 4 5

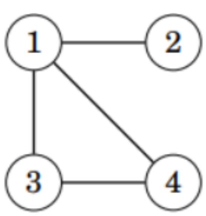
13.3 Total Number of Spanning Trees in a Graph

If a graph is a complete graph with n vertices, then total number of spanning trees is $n^{(n-2)}$ where n is the number of nodes in the graph. In complete graph, the task is equal to counting different labeled trees with n nodes for which have Cayley's formula.

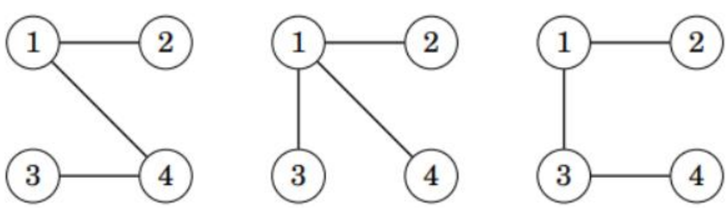
Laplacian matrix:

A Laplacian matrix L, where L[i, i] is the degree of node i and L[i, j] = −1 if there is an edge between nodes i and j, and otherwise L[i, j] = 0.

Kirchhoff's theorem provides a way to calculate the number of spanning trees for a given graph as a determinant of a special matrix. Consider the following graph,



All possible spanning trees are as follows:



In order to calculate the number of spanning trees, construct a Laplacian matrix L, where L[i, i] is the degree of node i and L[i, j] = −1 if there is an edge between nodes i and j, and otherwise L[i, j] = 0.
for the above graph, The Laplacian matrix will look like this

$$L = \begin{bmatrix} 3 & -1 & -1 & -1 \\ -1 & 1 & 0 & 0 \\ -1 & 0 & 2 & -1 \\ -1 & 0 & -1 & 2 \end{bmatrix}$$

The number of spanning trees equals the determinant of a matrix.

The Determinant of a matrix that can be obtained when we remove any row and any column from L.
For example, if we remove the first row and column, the result will be,

$$\det(\begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}) = 3.$$

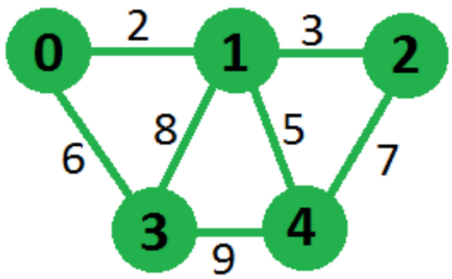
The determinant is always the same, regardless of which row and column we remove from L.

```
# Finds the number of spanning trees in a graph using Matrix Chain Multiplication.
MAX = 100
MOD = 1000000007

# Matrix Multiplication
def multiply(A, B, C):
    # write your code here
    …

# Function to find Nth power of A
def power(A, N, result):
    # write your code here
    …
# Function to find number of Spanning Trees in a Graph
# using Matrix Chain Multiplication.
def numOfSpanningTree(graph, V):
    # write your code here
    …

# Driver program
V = 4 # Number of vertices in graph
E = 5 # Number of edges in graph
graph = [[0, 1, 1, 1],
         [1, 0, 1, 1],
         [1, 1, 0, 1],
         [1, 1, 1, 0]]
print(numOfSpanningTree(graph, V))
```

13.4 Minimum Product Spanning Tree

A minimum product spanning tree for a weighted, connected, and undirected graph is a spanning tree with a weight product less than or equal to the weight product of every other spanning tree. The weight product of a spanning tree is the product of weights corresponding to each edge of the spanning tree. All weights of the given graph will be positive for simplicity.

Input:



Output: Minimum Product that we can obtain is 180 for above graph by choosing edges 0-1, 1-2, 0-3 and 1-4

This problem can be solved using standard minimum spanning tree algorithms like Kruskal and prim's algorithm, but we need to modify our graph to use these algorithms. Minimum spanning tree algorithms tries to minimize the total sum of weights, here we need to minimize the total product of weights. We can use the property of logarithms to overcome this problem.

log(w1* w2 * w3 * .... * wN) = log(w1) + log(w2) + log(w3) ..... + log(wN)

We can replace each weight of the graph by its log value, then we apply any minimum spanning tree algorithm which will try to minimize the sum of log(wi) which in turn minimizes the weight product.

```
# Minimum product spanning tree
import math

# Number of vertices in the graph
V = 5

# A utility function to find the vertex with minimum key value, from the set
# of vertices not yet included in MST
def minKey(key, mstSet):
    # write your code here
    …

# A utility function to print the constructed MST stored in parent[] and
# print Minimum Obtainable product
def printMST(parent, n, graph):
    # write your code here
    …

# Function to construct and print MST for a graph represented using adjacency
# matrix representation inputGraph is sent for printing actual edges and
# logGraph is sent for actual MST operations
def primMST(inputGraph, logGraph):
    # write your code here
    …

# Method to get minimum product spanning tree
def minimumProductMST(graph):
    # write your code here
    …

# Driver code
```

```
graph = [ [ 0, 2, 0, 6, 0 ],
          [ 2, 0, 3, 8, 5 ],
          [ 0, 3, 0, 0, 7 ],
          [ 6, 8, 0, 0, 9 ],
          [ 0, 5, 7, 9, 0 ], ]

# Print the solution
minimumProductMST(graph)
```

13.5 Reverse Delete Algorithm for Minimum Spanning Tree

In Reverse Delete algorithm, we sort all edges in decreasing order of their weights. After sorting, we one by one pick edges in decreasing order. We include current picked edge if excluding current edge causes disconnection in current graph. The main idea is delete edge if its deletion does not lead to disconnection of graph.

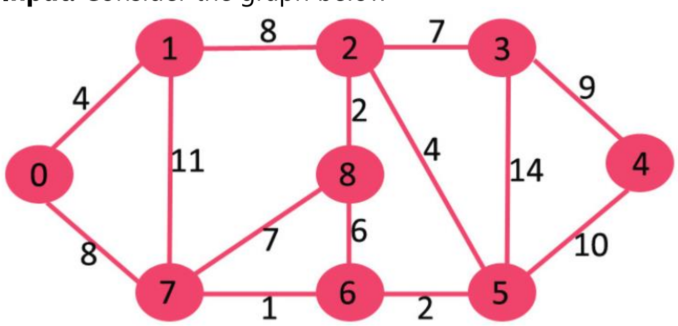
Algorithm:

1. Sort all edges of graph in non-increasing order of edge weights.

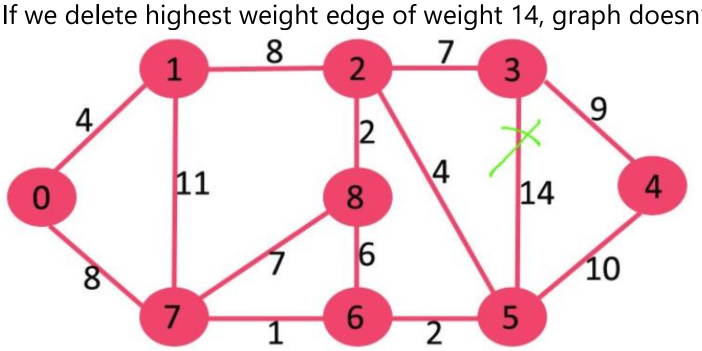
2. Initialize MST as original graph and remove extra edges using step 3.

3. Pick highest weight edge from remaining edges and check if deleting the edge disconnects the graph or not.
   If disconnects, then we don't delete the edge.
   Else we delete the edge and continue.

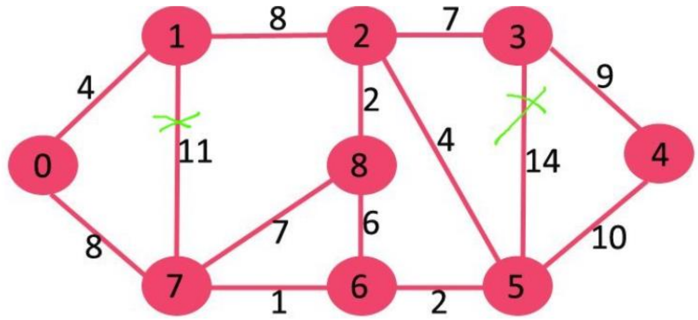
Input: Consider the graph below



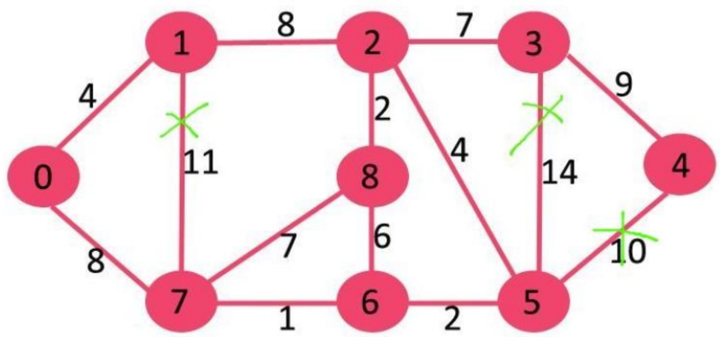
If we delete highest weight edge of weight 14, graph doesn't become disconnected, so we remove it.



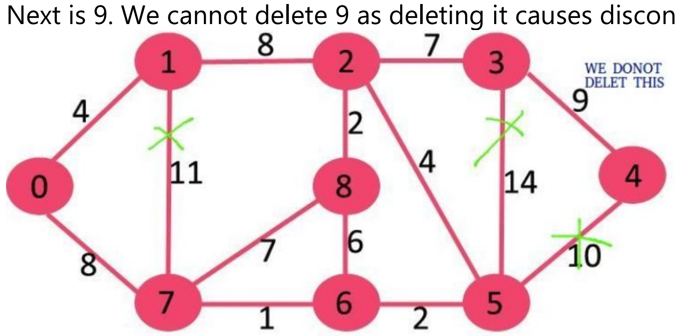
Next we delete 11 as deleting it doesn't disconnect the graph.



Next we delete 10 as deleting it doesn't disconnect the graph.



Next is 9. We cannot delete 9 as deleting it causes disconnection.



We continue this way and following edges remain in final MST.

Edges in MST

(3, 4)
(0, 7)
(2, 3)
(2, 5)
(0, 1)
(5, 6)
(2, 8)
(6, 7)

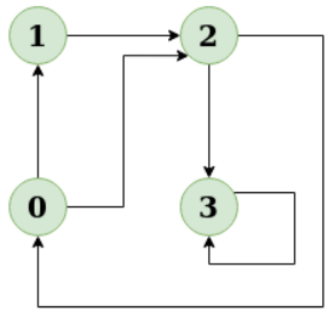
```
# Find Minimum Spanning Tree of a graph using Reverse Delete Algorithm

# Graph class represents a directed graph using adjacency list representation
class Graph:
```

```python
    def __init__(self, v):

        # No. of vertices
        self.v = v
        self.adj = [0] * v
        self.edges = []
        for i in range(v):
            self.adj[i] = []

    # function to add an edge to graph
    def addEdge(self, u: int, v: int, w: int):
        # write code here
        …

    def dfs(self, v: int, visited: list):
        # write code here
        …

    # Returns true if graph is connected
    # Returns true if given graph is connected, else false
    def connected(self):
        # write code here
        …

    # This function assumes that edge (u, v) exists in graph or not
    def reverseDeleteMST(self):
        # write code here
        …

# Driver Code
# create the graph given in above figure
V = 9
g = Graph(V)

# making above shown graph
g.addEdge(0, 1, 4)
g.addEdge(0, 7, 8)
g.addEdge(1, 2, 8)
g.addEdge(1, 7, 11)
g.addEdge(2, 3, 7)
g.addEdge(2, 8, 2)
g.addEdge(2, 5, 4)
g.addEdge(3, 4, 9)
g.addEdge(3, 5, 14)
g.addEdge(4, 5, 10)
g.addEdge(5, 6, 2)
g.addEdge(6, 7, 1)
g.addEdge(6, 8, 6)
g.addEdge(7, 8, 7)

g.reverseDeleteMST()
```

Try:

1. **Detect Cycle in a Directed Graph:** Given the root of a Directed graph, The task is to check whether the graph contains a cycle or not.

Input: N = 4, E = 6



Output: Yes
Explanation: The diagram clearly shows a cycle 0 -> 2 -> 0

14. Final Notes

The only way to learn programming is program, program and program on challenging problems. The problems in this tutorial are certainly NOT challenging. There are tens of thousands of challenging problems available – used in training for various programming contests (such as International Collegiate Programming Contest (ICPC), International Olympiad in Informatics (IOI)). Check out these sites:

- The ACM - ICPC International collegiate programming contest (https://icpc.global/)
- The Topcoder Open (TCO) annual programming and design contest (https://www.topcoder.com/)
- Universidad de Valladolid's online judge (https://uva.onlinejudge.org/).
- Peking University's online judge (http://poj.org/).
- USA Computing Olympiad (USACO) Training Program @ http://train.usaco.org/usacogate.
- Google's coding competitions (https://codingcompetitions.withgoogle.com/codejam, https://codingcompetitions.withgoogle.com/hashcode)
- The ICFP programming contest (https://www.icfpconference.org/)
- BME International 24-hours programming contest (https://www.challenge24.org/)
- The International Obfuscated C Code Contest (https://www0.us.ioccc.org/main.html)
- Internet Problem Solving Contest (https://ipsc.ksp.sk/)
- Microsoft Imagine Cup (https://imaginecup.microsoft.com/en-us)
- Hewlett Packard Enterprise (HPE) Codewars (https://hpecodewars.org/)
- OpenChallenge (https://www.openchallenge.org/)

Coding Contests Scores

Students must solve problems and attain scores in the following coding contests:

Name of the contest	Minimum number of problems to solve	Required score
• CodeChef	20	200
• Leetcode	20	200
• GeeksforGeeks	20	200
• SPOJ	5	50
• InterviewBit	10	1000
• Hackerrank	25	250

• Codeforces	10	100
• BuildIT	50	500
	Total score need to obtain	2500

Student must have any one of the following certifications:

1. HackerRank - Problem Solving Skills Certification (Basic and Intermediate)
2. GeeksforGeeks – Data Structures and Algorithms Certification
3. CodeChef - Learn Data Structures and Algorithms Certification
4. Interviewbit – DSA pro / Python pro
5. Edx – Data Structures and Algorithms
5. NPTEL – Programming, Data Structures and Algorithms
6. NPTEL – Introduction to Data Structures and Algorithms
7. NPTEL – Data Structures and Algorithms
8. NPTEL – Programming and Data Structure

V. TEXT BOOKS:

1. Rance D. Necaise, “Data Structures and Algorithms using Python”, Wiley Student Edition.
2. Benjamin Baka, David Julian, “Python Data Structures and Algorithms”, Packt Publishers, 2017.

VI. REFERENCE BOOKS:

1. S. Lipschutz, “Data Structures”, Tata McGraw Hill Education, 1st edition, 2008.
2. D. Samanta, “Classic Data Structures”, PHI Learning, 2nd edition, 2004.

VII. ELECTRONICS RESOURCES:

1. https://www.tutorialspoint.com/data_structures_algorithms/algorithms_basics.htm
2. https://www.codechef.com/certification/data-structures-and-algorithms/prepare
3. https://www.cs.auckland.ac.nz/software/AlgAnim/dsToC.html
4. https://online-learning.harvard.edu/course/data-structures-and-algorithms

VIII. MATERIALS ONLINE

1. Course Template
2. Lab manual

ESSENCE OF INDIAN TRADITIONAL KNOWLEDGE

III Semester: Common for all branches									
Course Code	Category	Hours / Week			Credits	Maximum Marks		
AHSC10	Mandatory	L	T	P	C	CIA	SEE	Total
		-	-	-	-	-	-	-
Contact Classes: Nil	Tutorial Classes: Nil	Practical Classes: Nil				Total Classes: Nil		
Prerequisite: No Prerequisites								

COURSE OBJECTIVES:

The course should enable the students to:

I. Understand the concept of Traditional knowledge and its importance
II. Know the need and importance of protecting traditional knowledge.
III. Know the various enactments related to the protection of traditional knowledge.
IV. Understand the concepts of Intellectual property to protect the traditional knowledge

MODULE-I INTRODUCTION TO TRADITIONAL KNOWLEDGE

Define traditional knowledge, nature and characteristics, scope and importance, kinds of traditional knowledge, the physical and social contexts in which traditional knowledge develop, the historical impact of social change on traditional knowledge systems. Indigenous Knowledge (IK), characteristics, traditional knowledge vis-à-vis indigenous knowledge, traditional knowledge Vs western knowledge traditional knowledge vis-à-vis formal knowledge

MODULE-II PROTECTION OF TRADITIONAL KNOWLEDGE

Protection of traditional knowledge: The need for protecting traditional knowledge Significance of TK Protection, value of TK in global economy, Role of Government to harness TK.

MODULE-III LEGAL FRAMEWORK AND TK

A: The Scheduled Tribes and Other Traditional Forest Dwellers (Recognition of Forest Rights) Act, 2006, Plant Varieties Protection and Farmer's Rights Act, 2001 (PPVFR Act);

B: The Biological Diversity Act 2002 and Rules 2004, the protection of traditional knowledge bill, 2016. Geographical indicators act 2003.

MODULE-IV TRADITIONAL KNOWLEDGE AND INTELLECTUAL PROPERTY

Systems of traditional knowledge protection, Legal concepts for the protection of traditional knowledge, Certain non IPR mechanisms of traditional knowledge protection, Patents and traditional knowledge, Strategies to increase protection of traditional knowledge, global legal FORA for increasing protection of Indian Traditional Knowledge.

MODULE-V TRADITIONAL KNOWLEDGE IN DIFFERENT SECTORS:

Traditional knowledge and engineering, Traditional medicine system, TK and biotechnology, TK in agriculture, Traditional societies depend on it for their food and healthcare needs, Importance of conservation and sustainable development of environment, Management of biodiversity, Food security of the country and protection of TK. 139.

Text Books:

1. Traditional Knowledge System in India, by Amit Jha, 2009.
2. Traditional Knowledge System and Technology in India by Basanta Kumar Mohanta and Vipin Kumar Singh Pratibha Prakashan 2012.

Reference Books:

1. Traditional Knowledge System in India by Amit Jha Atlantic publishers, 2002.
2. "Knowledge Traditions and Practices of India" Kapil Kapoor1, Michel Danino2

COMPLEX ANALYSIS AND PROBABILITY DISTRIBUTIONS

<table>
<tr><td colspan="10">IV Semester: EEE</td></tr>
<tr><th>Course Code</th><th>Category</th><th colspan="3">Hours / Week</th><th>Credits</th><th colspan="3">Maximum Marks</th></tr>
<tr><td rowspan="2">AHSC11</td><td rowspan="2">Foundation</td><td>L</td><td>T</td><td>P</td><td>C</td><td>CIA</td><td>SEE</td><td>Total</td></tr>
<tr><td>3</td><td>1</td><td>-</td><td>4</td><td>30</td><td>70</td><td>100</td></tr>
<tr><td>Contact Classes: 45</td><td>Tutorial Classes: 15</td><td colspan="4">Practical Classes: Nil</td><td colspan="3">Total Classes: 60</td></tr>
<tr><td colspan="10">Prerequisite: Linear Algebra and Calculus, Mathematical Transform Techniques</td></tr>
</table>

I. COURSE OVERVIEW:

This course Complex Analysis and Probability Distributions provides an introduction to complex analysis which is theory of complex functions with complex variable and random variables. The course includes complex functions and differentiation, complex integration, power series expansion of complex function, single random variables and probability distributions. The mathematical skills derived from this course form a necessary base to analytical and design concepts encountered in the program.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The applications of complex variable and conformal mapping in two dimensional complex potential theories.
II. The fundamental calculus theorems and criteria for the independent path on contour integral used in problems of engineering.
III. The concepts of probability on single random variables and probability distributions.
IV. The theory of random variables, basic random variate distributions and complex analysis for understanding the numerical growth rates.

III. COURSE SYLLABUS:

MODULE-I: COMPLEX FUNCTIONS AND DIFFERENTIATION (09)

Complex functions differentiation and integration: Complex functions and its representation on argand plane, concepts of limit, continuity, differentiability, analyticity, Cauchy-Riemann conditions and harmonic functions; Milne-Thomson method. Bilinear Transformation.

MODULE-II: COMPLEX INTEGRATION (09)

Line integral: Evaluation along a path and by indefinite integration; Cauchy's integral theorem; Cauchy's integral formula; Generalized integral formula; Power series expansions of complex functions and contourIntegration: Radius of convergence.

MODULE-III: POWER SERIES EXPANSION OF COMPLEX FUNCTION (09)

Expansion in Taylor's series, Maclaurin's series and Laurent series. Singular point; Isolated singular point; Pole of order m; Essential singularity; Residue: Cauchy Residue Theorem.

Evaluation of Residue by Laurent Series and Residue Theorem. Evaluation of integrals of the type $\int_0^{2\pi} f(cos\theta, sin\theta)d\theta, \int_{-\infty}^{\infty} f(x)dx$

MODULE-IV: SINGLE RANDOM VARIABLES (09)

Random variables: Discrete and continuous, probability distributions, mass function-density function of a probability distribution. Mathematical expectation, moment about origin, central moments, moment generating function of probability distribution.

MODULE-V: PROBABILITY DISTRIBUTIONS (09)

Binomial, Poisson and normal distributions and their properties.

IV. TEXT BOOKS:

1. Erwin Kreyszig, "Advanced Engineering Mathematics", John Wiley & Sons Publishers, 10^{th} Edition, 2014.

2. B S Grewal, “Higher Engineering Mathematics”, Khanna Publishers, 42nd Edition, 2012.

V. REFERENCE BOOKS:

1. Churchill, RV and Brown, J W, “Complex Variables and Applications”, Tata McGraw-Hill, 8th Edition, 2012.
2. A K Kapoor, “Complex Variables Principles and Problem Sessions”, World Scientific Publishers, 1st Edition, 2011.
3. Murray Spiegel, John Schiller, “Probability and Statistics”, Schaum‟s Outline Series, 3rd Edition, 2010.

VI. WEB REFERENCES:

1. http://www.efunda.com/math/math_home/math.cfm
2. http://www.ocw.mit.edu/resourcs/#Mathematics
3. http://www.sosmath.com
4. http://www.mathworld.wolfram.com

VII. E-TEXT BOOKS:

1. http://www.keralatechnologicaluniversity.blogspot.in/2015/06/erwin-kreyszig-advanced-engineering-mathematics-ktu-ebook-download.html
2. http://www.faadooengineers.com/threads/13449-Engineering-Maths-II-eBooks

ELECTRICAL POWER GENERATION SYSTEMS

IV Semester: EEE								
Course Code	Category	Hours / Week			Credits	Maximum Marks		
AEEC10	Core	L	T	P	C	CIA	SEE	Total
		3	1	0	4	30	70	100
Contact Classes: 45	Tutorial Classes: 15	Practical Classes: Nil				Total Classes: 60		
Prerequisite: Electrical Circuits (AEEC02), Electromagnetic Fields ().								

I. COURSE OVERVIEW:

This course provides ability to recognize, analyze and troubleshoot different elements in electric power generation systems. It deals with conventional energy systems like thermal and nuclear power stations. This course also introduces non-conventional energy systems like solar energy (radiation, collection, storage, and application), Hydro and Wind energy. This course will also discuss some environmental impacts of power generation and also look at alternative and sustainable energy resources**.**

II. COURSE OBJECTIVES:

The students will try to learn:

I. Demonstrate various conventional power generation systems including major subsystems.
II. Understand hydroelectric power generation systems along with pumped storage plants and hydraulic turbines.
III. Apply knowledge of solar and wind power generation systems in design and implementation to obtain clean energy.
IV. Illustrate the economic aspects of power generation and power tariff methods.

III. COURSE SYLLABUS:

MODULE-I: CONVENTIONAL POWER GENERATION SYSTEMS (09)

Thermal Power Stations: Evaluation of power systems, present day scenario, Line diagram of thermal power station (TPS) showing paths of coal, steam, water, air, ash and flue gasses; Brief description of TPS components: Economizers, boilers, super heaters, turbines, condensers, chimney and cooling towers. Nuclear power stations: Nuclear fission and chain reaction, nuclear fuels, principle of operation of nuclear reactor, reactor components, moderators, control rods, reflectors and coolants, radiation hazards, shielding and safety precautions, types of nuclear reactors and brief description of PWR, BWR and FBR; Gas power stations: Principle of operation and components (Block diagram approach only).

MODULE-II: HYDROELECTRIC POWER STATIONS (09)

Hydroelectric Power Stations: Elements of hydro electric power station, types, concept of pumped storage plants, storage requirements, mass curve (explanation only), estimation of power developed from a given catchment area, heads and efficiencies; Hydraulic turbines: Classification of turbines, impulse and reaction turbines, Pelton wheel, Francis turbine and Kaplan turbine, working proportions, work done, efficiencies, hydraulic design, draft tube theory, functions and efficiency.

MODULE-III: SOLAR ENERGY (09)

Solar radiation: Environmental impact of solar power, physics of the sun, solar constant, extraterrestrial and terrestrial solar radiation, solar radiation on tilted surface, instruments for measuring solar radiation, solar radiation data, solar concentrators, collectors, thermal applications, design of standalone solar systems, simple problems.

Photovoltaic systems: Photovoltaic effect, semiconducting materials, band gap theory, photo emission of electrons, cell configuration, types of solar cells, cell properties, device physics, electrostatic field across the depletion layer, voltage developed, I-V characteristics, module structure and fabrication, output power and efficiency, fill factor, maximum power point tracking (MPPT), solar grid connected inverters, simple problems.

MODULE-IV: WIND ENERGY (09)

Wind energy: Sources and potential, power from wind, Betz criterion, components of wind energy conversion system, types of turbines, horizontal and vertical axis wind turbines, aerodynamics, momentum theory (actuator disk concept), operational characteristics, blade element theory, types of generating systems for wind energy, permanent magnet generators, DC generators, induction generators, doubly fed induction generators, applications of wind energy, safety and environmental aspects, simple problems.

MODULE-V: ECONOMIC ASPECTS OF POWER GENERATION (09)

Terms commonly used in system operation, various factors affecting cost of generations; load curves, connected load, maximum demand, peak load, base load and peak load power plants, load factors, plant capacity factor, plant use factor, demand factors, diversity factor, cost of power plant, tariffs.

IV. TEXT BOOKS:

1. C L Wadhawa, "Generation, Distribution and Utilization of Electrical Energy", New Age International Limited, New Delhi, 3rd Edition, 2005.
2. G D Rai, "Non-Conventional Energy Sources", Khanna Publishers, 1st Edition, 2011.
3. G N Tiwari, M K Ghosal, "Fundamentals of Renewable Energy Sources", Narosa Publications, New Delhi, 1st Edition, 2007.
4. Chetan Singh Solanki, "Solar Photovoltaics", PHI Publications, 2nd Edition, 2011.
5. M L Soni, P V Gupta, U S Bhatnagar and A Chakraborti, "A text book on Power system engineering", Dhanpat Rai and Co. Pvt. Ltd, 1999.

V. REFERENCE BOOKS:

1. J B Gupta, "A Course in Electrical Power", S K Kataria and Sons, New Delhi, 15th Edition, 2013.
2. M V Deshpande, "Elements of Power Station design", Prentice Hall India Learning Private Limited, New Delhi, 1st Edition, 1992.
3. Mukund R Patel, "Wind and Solar Power Systems", CRC Press, 1st Edition, 1999.
4. V K Mehta and Rohit Mehta, "Principle of Power Systems", S Chand & Company, Ltd, New Delhi, 3rd Edition, 2005.

VI. WEB REFERENCES:

1. https://www.solarpowernotes.com
2. https://www.electrical4u.com/power-plants-types-of-power-plant
3. https://www.iare.ac.in

VII. E-TEXT BOOKS:

1. https://www.amazon.in/Electrical-Power-Engineering-Reference-Applications
2. https://www.nitt.edu
3. https://www.textbooksonline.tn.nic.in

AC MACHINES

<table>
<tr><td colspan="9">IV Semester: EEE</td></tr>
<tr><th>Course Code</th><th>Category</th><th colspan="3">Hours / Week</th><th>Credits</th><th colspan="3">Maximum Marks</th></tr>
<tr><td rowspan="2">AEEC11</td><td rowspan="2">Core</td><td>L</td><td>T</td><td>P</td><td>C</td><td>CIA</td><td>SEE</td><td>Total</td></tr>
<tr><td>3</td><td>0</td><td>0</td><td>3</td><td>30</td><td>70</td><td>100</td></tr>
<tr><td>Contact Classes: 45</td><td>Tutorial Classes: Nil</td><td colspan="4">Practical Classes: Nil</td><td colspan="3">Total Classes: 45</td></tr>
<tr><td colspan="9">Prerequisite: Linear Algebra and Calculus, DC Machines and Transformers</td></tr>
</table>

I. COURSE OVERVIEW:

This course deals with the basic theory, construction, operation, performance characteristics and application of electromechanical energy conversion devices such as synchronous and asynchronous machines. It also facilitates the study of the alternating machines which are the major part of industrial drives and agricultural pump sets.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The principle of operation and the effect of pulsating, rotating magnetic fields on the working of AC machines

II. The armature winding layouts and concept of armature reaction with phasor diagrams.

III. The staring, speed control methods and equivalent circuit diagram of poly phase and single phase machines.

III. COURSE SYLLABUS:

MODULE-I: THREE PHASE INDUCTION MACHINES (09)

Magnetic fields: Constant magnetic field, pulsating magnetic field, rotating magnetic field; Three phase induction motors: Construction, types of induction motors, slip and frequency of rotor currents, rotor MMF and production of torque, equivalent circuit, power across air gap, torque and power output, torque slip characteristics, generating and braking modes, maximum (breakdown) torque, starting torque, maximum power output. Equivalent circuit; Induction generator: Operation, approximate equivalent circuit, doubly fed induction generator, numerical problems.

MODULE-II: TESTING AND SPEED CONTROL OF INDUCTION MOTORS (09)

Testing: Brake test, no load and blocked rotor test, determination of induction motor parameters from circle diagram, numerical problems. Starting methods of Slip ring and squirrel cage induction motors; Speed control of induction motors, numerical problems.

MODULE-III: ALTERNATORS (09)

Synchronous generators: Introduction, principle of operation, types, constructional features, integral slot and fractional slot windings, distributed and concentrated windings, winding factors, basic synchronous machine model, circuit model of a synchronous machine, armature reaction, phasor diagrams.

Voltage regulation: Determination of synchronous impedance, short circuit ratio, leakage reactance, calculation of regulation by synchronous impedance method, MMF, ZPF and ASA methods; Parallel operation, synchronization of alternators; Slip test, numerical problems.

MODULE-IV: SYNCHRONOUS MOTORS (09)

Synchronous motors: Principle of operation, phasor diagrams, power developed, synchronous motor with different excitations, effect of increased load with constant excitation, effect of change in excitation with constant load, effect of excitation on armature current and power factor, construction of “V” and inverted “V” curves, power and excitation circles, starting methods, and analysis, synchronous condenser.

MODULE-V: SINGLE-PHASE INDUCTION MOTORS (09)

Single phase induction motor: Principle of operation, two reaction theory, equivalent circuit analysis, split phase motor, construction, principle of operation, capacitor start, capacitor run, capacitor start - capacitor run motor, shaded pole motor, torque speed characteristics of single phase induction motors.

IV. TEXT BOOKS:

1. P S Bimbhra, "Electrical Machinery", Khanna Publishers, 1st Edition, 2011.
2. I J Nagrath and D P Kothari, "Electric Machines", McGraw Hill Education, 1st Edition, 2010.
3. J B Guptha "Theory and performance of Electrical machiines", S.K.Kataria & Sons Publishers 14th Edition, 2009.

V. REFERENCE BOOKS:

1. M G Say, E O Taylor, "Direct Current Machines", Longman Higher Education, 1st Edition, 1985.
2. M G Say, "Performance and design of AC machines", CBS Publishers, 1st Edition, 2002.
3. A E Fitzgerald and C Kingsley, "Electric Machinery", New York, McGraw Hill Education, 1st Edition, 2013
4. M V Deshpande, "Electrical Machines", PHI Learning Private Limited, 3rd Edition, 2011.

VI. WEB REFERENCES:

1. https://www.electrical4u.com
2. https://www.freevideolectures.com

VII. E-TEXT BOOKS:

1. https://www.freeengineeringbooks.com
2. https://www.pdfdrive.com/textbook-of-electrical-technology-ac-and-dc-machines-d184089760.html

CONTROL SYSTEMS

IV Semester: EEE								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
AEEC12	**Core**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		3	0	0	3	30	70	100
Contact Classes: 45	**Tutorial Classes: Nil**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite: Linear Algebra and Calculus (AHSC02), DC Machines and Transformers()								

I. COURSE OVERVIEW:

This course deals with the basic concepts of block diagram reduction technique, time response analysis of first order and second order systems. It deals with various time and frequency domain analysis. It elaborates the concept of stability and its assessment for linear time invariant systems. This course address the various real time issues and how the control strategies are used in automation areas associates with variety of engineering streams

II. COURSE OBJECTIVES:

The students will try to learn:

I. Organize modeling and analysis of electrical and mechanical systems.
II. Analyze control systems by block diagrams and signal flow graph technique.
III. Demonstrate the analytical and graphical techniques to study the stability.
IV. Illustrate the frequency domain and state space analysis.

III. COURSE SYLLABUS:

MODULE-I: INTRODUCTION AND MODELING OF PHYSICAL SYSTEMS (08)

Control systems: Introduction, open loop and closed loop systems, examples, comparison, mathematical modeling and differential equations of physical systems, concept of transfer function, translational and rotational mechanical systems, electrical systems, force - voltage and force - current analogy.

MODULE-II: BLOCK DIAGRAM REDUCTION AND TIME RESPONSE ANALYSIS (10)

Block Diagrams: Block diagram representation of various systems, block diagram algebra, characteristics of feedback systems, AC servomotor, signal flow graph, Mason's gain formula; Time response analysis: Standard test signals, shifted unit step, impulse response, unit step response of first and second order systems, time response specifications, steady state errors and error constants, dynamic error coefficients method, effects of P, PD, PI and PID controllers.

MODULE-III: CONCEPT OF STABILITY AND ROOT LOCUS TECHNIQUE (09)

Concept of stability: Necessary and sufficient conditions for stability, Routh's and Routh Hurwitz stability criterions and limitations.

Root locus technique: Introduction, root locus concept, construction of root loci, graphical determination of 'k' for specified damping ratio, relative stability, effect of adding zeros and poles on stability.

MODULE-IV: FREQUENCY DOMAIN ANALYSIS (10)

Frequency domain analysis: Introduction, frequency domain specifications, stability analysis from Bode plot, Nyquist plot, calculation of gain margin and phase margin, determination of transfer function, correlation between time and frequency responses.

MODULE-V: STATE SPACE ANALYSIS AND COMPENSATORS (08)

State Space Analysis: Concept of state, state variables and state model, derivation of state models from block diagrams, diagonalization, solving the time invariant state equations, state transition matrix and properties, concept of controllability and observability; Compensators: Lag, lead, lead - lag networks.

IV. TEXT BOOKS:

1. I J Nagrath, M Gopal, "Control Systems Engineering", New Age International Publications, 3rd Edition, 2007.
2. K Ogata, "Modern Control Engineering", Prentice Hall, 4th Edition, 2003.
3. N C Jagan, "Control Systems", BS Publications, 1st Edition, 2007.

V. REFERENCE BOOKS:

1. Anand Kumar, “Control Systems”, PHI Learning, 1st Edition, 2007.
2. S Palani, “Control Systems Engineering”, Tata McGraw-Hill Publications, 1st Edition, 2001.
3. N K Sinha, “Control Systems”, New Age International Publishers, 1st Edition, 2002.

VI. WEB REFERENCES:

1. https://www.researchgate.net
2. https://www.aar.faculty.asu.edu/classes
3. https://www.facstaff.bucknell.edu/
4. https://www.electrical4u.com
5. https://www.iare.ac.in

VII. E-TEXT BOOKS:

1. https://www.jntubook.com/
2. https://www.freeengineeringbooks.com

DIGITAL ELECTRONICS

IV Semester: EEE								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
AECC16	**Core**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		3	-	-	3	30	70	100
Contact Classes: 45	**Tutorial Classes: Nil**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite: There are no prerequisites to take this course								

I. COURSE OVERVIEW:

This course intended to logic gates, various logic families. Design of digital circuits using logic gates, combinational circuits and sequential circuits. Apply op-amp characteristics to design analog to digital converters and digital to analog converters. Classification and characteristics of memories such as Read-only memory, Random access memory and programmable logic devices such as programmable logic array and programmable array logic

II. COURSE OBJECTIVES:

The students will try to learn:

I. Demonstrate the concept of electrostatic field intensity and electric potential.
II. Illustrate polarization of dielectrics and the behavior of conductors and dielectrics in electric field.
III. Understand the concept of magnetic field intensity and flux density.
IV. Discuss forces in magnetic fields and law of electromagnetic induction.
V. Analyze propagation of electro-magnetic waves.

III. COURSE SYLLABUS:

MODULE-I: FUNDAMENTALS OF DIGITAL SYSTEMS AND LOGIC FAMILIES (09)

Digital signals, digital circuits, AND, OR, NOT, NAND, NOR and Exclusive-OR operations, Boolean algebra, examples of IC gates, number systems-binary, signed binary, octal hexadecimal number, binary arithmetic, one’s and two’s complements arithmetic, codes, error detecting and correcting codes, characteristics of digital lCs, digital logic families, TTL, Schottky TTL and CMOS logic, interfacing CMOS and TTL, Tri-state logic.

MODULE-II: COMBINATIONAL DIGITAL CIRCUITS (09)

Standard representation for logic functions, K-map representation, and simplification of logic functions using Kmap, minimization of logical functions. Don’t care conditions, Multiplexer, De-Multiplexer, Decoders, Adders, Sub tractors, BCD arithmetic, carry look ahead adder, serial ladder, ALU, elementary ALU design, popular MSI chips, digital comparator, parity checker/generator, code converters, priority encoders, decoders, drivers for display devices, Q-M method of function realization.

MODULE-III: SEQUENTIAL CIRCUITS AND SYSTEMS (09)

1-bit memory, the circuit properties of Bi-stable latch, the clocked SR flip flop, J- K-T and D types flip flops, applications of flip flops, shift registers, applications of shift registers.

Serial to parallel converter: Parallel to serial converter, ring counter, sequence generator, ripple (Asynchronous) counters, synchronous counters, counters design using flip flops, special counter IC’s, asynchronous sequential counters, applications of counters.

MODULE-IV: A/D AND D/A CONVERTERS (09)

Digital to analog converters: weighted resistor, converter, R-2R Ladder D/A converter, specifications for D/A converters, examples of D/A converter lCs, sample and hold circuit, analog to digital converters: quantization and encoding, parallel comparator A/D converter, successive approximation A/D converter, counting A/D converter, dual slope A/D converter, A/D converter using voltage to frequency and voltage to time conversion, specifications of A/D converters, example of A/D converter ICs.

MODULE-V: SEMICONDUCTOR MEMORIES AND PROGRAMMABLE LOGIC DEVICES (09)

Memory organization and operation, expanding memory size, classification and characteristics of memories, sequential memory, read only memory (ROM), read and write memory(RAM), content addressable memory (CAM), charge de coupled device memory (CCD), commonly used memory chips, ROM as a PLD, Programmable logic array, Programmable array logic, complex Programmable logic devices (CPLDS), Field Programmable Gate Array (FPGA).

IV. TEXT BOOKS:

1. P Jain, "Modern Digital Electronics", McGraw Hill Education, 2009.
2. M M Mano, "Digital logic and Computer design", Pearson Education India, 2016.

V. REFERENCE BOOKS:

1. A Kumar, "Fundamentals of Digital Circuits", Prentice Hall India, 2016.

VI. WEB REFERENCES:

1. https://www.calvin.edu/~pribeiro/courses/engr315/EMFT_Book.pdf
2. https://www.web.mit.edu/viz/EM/visualizations/coursenotes/modules/guide02.pdf
3. https://www.nptel.ac.in/courses/108106073/
4. https://www.iare.ac.in

VII. E-TEXT BOOKS:

1. https://www.*bookboon.com/en/electromagnetism-for-electronic-engineers*
2. https://www.books.google.co.in/books/.../Fundamentals of Electromagnetic Fields
3. https://www.aliexpress.com/item/EBOOK...Electromagnetic-Fields-2

EXPERIENTIAL ENGINEERING EDUCATION (EXEED) – FABRICATION / MODEL DEVELOPMENT

IV Semester: Common for all branches								
Course Code	Category	Hours / Week			Credits	Maximum Marks		
ACSC14	Foundation	L	T	P	C	CIA	SEE	Total
		2	0	0	1	30	70	100
Contact Classes: 28	Tutorial Classes: Nil	Practical Classes: Nil				Total Classes: 28		
Prerequisite: There are no prerequisites to take this course								

COURSE OVERVIEW:

This course provide the environment to develop high-tech, ecological and socially responsible products from concept and design to production. The course covers hands-on learning in product and industrial design and product development with product lifecycle management.

OBJECTIVES:

The students will try to learn:

I. The design thinking process and Identify opportunities through customer needs analysis.
II. Product specifications based on customer needs that are desirable, feasible, and viable through applied creativity.
III. The Implementation techniques for planning and executing a prototype design services.

WEEK NO	TOPIC
WEEK – I	Introduction To Product Design
WEEK – II	Design Thinking Skills
WEEK – III	Identifying Customer Needs
WEEK – IV	Product Specifications
WEEK – V	Applied Creativity
WEEK – VI	Prototyping
WEEK – VII	Design Of Services
WEEK –VIII	Product Architecture
WEEK - IX	Financial Analysis
WEEK - X	Design For Environment
WEEK - XI	Product Development Process
WEEK - XII	Reverse Engineering
WEEK - XIII	Value Engineering
WEEK - XIV	Assessment

AC MACHINES LABORATORY

IV Semester: EEE									
Course Code	Category	Hours / Week			Credits	Maximum Marks		
AEEC13	Core	L	T	P	C	CIA	SEE	Total
		0	0	3	1.5	30	70	100
Contact Classes: Nil	Tutorial Classes: Nil	Practical Classes: 36				Total Classes:36		
Prerequisite: Electrical Circuits Linear Algebra and Calculus								

I. COURSE OVERVIEW:

This is the main lab where experiments like load test on various machines, speed control tests, open circuit tests, short circuit tests, etc are carried out. And also wide variety of practical experiments are performed here with combination of different rotating machines. The laboratory is also used for research activities in machines and to carry out project works on energy conversion.

II. COURSE OBJECTIVES:

The students will try to learn:

I. Evaluate losses and determine the efficiency of single phase and three phase electrical machines.
II. Determine the voltage regulation, efficiency and temperature rise in various transformers.
III. Apply PLC and digital simulation software to gain practical knowledge.

III. COURSE SYLLABUS:

Week – 1: OC AND SC TEST ON SINGLE PHASE TRANSFORMER

Determine the equivalent circuit parameters; predetermine the efficiency and regulation by open circuit and short circuit test on a single phase transformer

Week – 2: SUMPNER'S TEST

Predetermine the efficiency and regulation of two identical single phase transformers

Week – 3: LOAD TEST ON SINGLE PHASE TRANSFORMERS

Determination of efficiency by load test on a single phase transformer

Week – 4: SCOTT CONNECTION OF TRANSFORMERS

Conversion of three phase to two phase using single phase transformers

Week – 5: SEPARATION OF CORE LOSSES IN SINGLE PHASE TRANSFORMER

Find out the eddy current and hysteresis losses in single phase transformer

Week – 6: HEAT RUN TEST ON SINGLE PHASE TRANSFORMERS

Determine the temperature rise in three single phase transformers set

Week – 7: BRAKE TEST ON THREE PHASE SQUIRREL CAGE INDUCTION MOTOR

Plot the performance characteristics of three phase induction motor

Week – 8: CIRCLE DIAGRAM OF THREE PHASE SQUIRREL CAGE INDUCTION MOTOR

Plot the circle diagram and predetermine the efficiency and losses of three phase squirrel cage induction motor

Week – 9: REGULATION OF ALTERNATOR BY EMF METHOD

Determine the regulation of alternator using synchronous impedance method

Week – 10: REGULATION OF ALTERNATOR BY MMF METHOD

Determine the regulation of alternator using amperes turns method

Week – 11: SLIP TEST ON THREE PHASE SALIENT POLE SYNCHRONOUS MOTOR
Determination of Xd and Xq in a three phase salient pole synchronous motor.

Week – 12: V' AND INVERTED 'V' CURVES OF SYNCHRONOUS MOTOR
Plot 'V' and inverted 'V' curves to study the effect of power factor in synchronous motor.

Week – 13: EQUIVALENT CIRCUIT PARAMETERS OF SINGLE PHASE INDUCTION MOTOR
Determine the equivalent circuit parameters of a single phase induction motor

Week – 14: STARTING AND SPEED CONTROL OF INDUCTION MOTOR USING PLC
Implementation of star-delta starter using PLC; Speed control of three phase slip ring induction motor with rotor resistance cutting using PLC

IV. REFERENCE BOOKS:

1. P S Bimbhra, "Electrical Machines", Khanna Publishers, 2nd Edition, 2008.
2. M G Say, E O Taylor, "Direct Current Machines", Longman Higher Education, 1st Edition, 1985.
3. Hughes, "Electrical Technology", Prentice Hall, 10th Edition, 2015.
4. Nesimi Ertugrul, "LabVIEW for Electric Circuits, Machines, Drives, and Laboratories", Prentice Hall, 1st Edition, 2002.
5. Gupta, Gupta & John, "Virtual Instrumentation Using LabVIEW", Tata McGraw-Hill, 1st Edition, 2005.

V. WEB REFERENCES:

1. https://www.ee.iitkgp.ac.in
2. https://www.citchennai.edu.in
3. https://www.iare.ac.in /

CONTROL SYSTEMS LABORATORY

IV Semester: EEE								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
AEEC14	**Core**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		0	0	3	1.5	30	70	100
Contact Classes: Nil	**Tutorial Classes: Nil**	**Practical Classes: 36**				**Total Classes:36**		
Prerequisite: Linear Algebra and Calculus (AHSC02), DC Machines and Transformers ().								

I. COURSE OVERVIEW:

Control Systems Laboratory presents facilities of computing and simulation through MATLAB and demonstration on FEEDBACK designed equipment with PCI cards creating an impressive digital control system development environment.., which can later be implemented in real-time applications using Real-time Workshop.

II. COURSE OBJECTIVES:

The students will try to learn:

I. Understand mathematical models of electrical and mechanical systems.
II. Analysis of control system stability using digital simulation.
III. Demonstrate the time domain and frequency domain analysis for linear time invariant systems
IV. Apply programmable logic controllers to demonstrate industrial controls in the laboratory.

III. COURSE SYLLABUS:

Week – 1: TIME RESPONSE OF SECOND ORDER SYSTEM

To obtain the time response of a given second order system with time domain specifications

Week – 2: TRANSFER FUNCTION OF DC MOTOR

Determine the transfer function, time response of DC motor and verification with digital simulation

Week – 3: AC SERVO MOTOR

Study of AC servomotor and plot its torque speed characteristics

Week – 4: EFFECT OF VARIOUS CONTROLLERS ON SECOND ORDER SYSTEM

Study the effect of P, PD, PI and PID controller on closed loop second order systems

Week – 5: COMPENSATOR

Study lead-lag compensator and obtain its magnitude, phase plots

Week – 6: TEMPERATURE CONTROLLER

Study the performance of PID controller used to control the temperature of an oven

Week – 7: DESIGN AND VERIFICATION OF OP-AMP BASED PID CONTROLLER

Implementation of PID controller using Op-Amps and verification using MATLAB

Week – 8: STABILITY ANALYSIS USING DIGITAL SIMULATION

Stability analysis using root locus, Bode plot, Polar, Nyquist criterions of linear time invariant system by digital simulation

Week – 9: STATE SPACE MODEL USING DIGITAL SIMULATION

Verification of state space model from transfer function and transfer function from state space model using digital simulation

Week – 10: LADDER DIAGRAMS USING PLC

Input output connection, simple programming, ladder diagrams, uploading, running the program and debugging in programmable logic controller

Week – 11: TRUTH TABLES USING PLC

Study and verification of truth tables of logic gates, simple boolean expressions and application to speed control of DC motor using programmable logic controller.

Week – 12: IMPLEMENTATION OF COUNTER

Implementation of counting number of objects and taking action using PLC.

Week – 13: BLINKING LIGHTS USING PLC

Implementation of blinking lights with programmable logic controller

Week – 14: WATER LEVEL CONTROL

Control of maximum and minimum level of water in a tank using PLC

IV. REFERENCE BOOKS:

1. J Nagrath, M Gopal, "Control Systems Engineering", New Age International, 3rd Edition, 2007.
2. K Ogata, "Modern Control Engineering", Prentice Hall, 4th Edition, 2003.
3. Benjamin Kuo, "Automatic Control Systems", PHI, 7th Edition, 1987.

V. WEB REFERENCES:

1. https://www.ee.iitkgp.ac.in
2. https://www.ggnindia.dronacharya.info/ece2dept/Downloads/Labmanuals/VI Sem/Control_ System _ Lab. pdf
3. https://www.iare.ac.in
4. https://www.deltaww.com

ANALOG AND DIGITAL ELECTRONICS LABORATORY

IV Semester: EEE								
Course Code	Category	Hours / Week			Credits	Maximum Marks		
AECC17	Core	L	T	P	C	CIA	SEE	Total
		0	0	2	1	30	70	100
Contact Classes: Nil	Tutorial Classes: Nil	Practical Classes: 36				Total Classes:36		
Prerequisite: There are no prerequisites to take this course.								

I. COURSE OVERVIEW:

The course has been designed to introduce fundamental principles of analog and digital electronics. The students completing this course will understand basic analog and digital electronics, including semiconductor properties, operational amplifiers, combinational and sequential logic and analog-to digital digital-to-analog conversion techniques. Finally, students will gain experience in with the design of analog amplifiers, power supplies and logic device

II. COURSE OBJECTIVES:

The students will try to learn:

I. Implement and study the characteristics of diodes and transistors.
II. Illustrate the concept of rectification using half wave and full wave rectifiers.
III. Design and construct different amplifier circuits.
IV. Build the concept of digital and binary system.
V. Design and analyze the combinational logic circuits.

III. COURSE SYLLABUS:

Week – 1: PN JUNCTION DIODE CHARACTERISTICS

Verification of V-I characteristics of PN diode and calculate static and dynamic resistance using Hardware.

Week – 2: ZENER DIODE CHARACTERISTICS AND VOLTAGE REGULATOR

Verification of V-I characteristics of Zener diode and perform Zener diode as a Voltage regulator using Hardware

Week – 3: HALF WAVE AND FULL WAVE RECTIFIER

Verification of Half wave rectifier and Full wave rectifier without and with filters using hardware

Week – 4: TRANSISTOR CE CHARACTERISTICS

Verification of Input and Output characteristics of CE configuration using hardware

Week – 5: TRANSISTOR CB CHARACTERISTICS

Verification of Input and Output characteristics of CB configuration using hardware

Week – 6: FREQUENCY RESPONSE OF CE AMPLIFIER

Determine the Gain and Bandwidth of CE amplifier using hardware

Week – 7: BOOLEAN EXPRESSIONS USING GATES

Realization of Boolean Expressions using Gates

Week – 8: UNIVERSAL GATES

Design and realization of logic gates using universal gates

Week – 9: NAND / NOR GATES

Generation of clock using NAND / NOR gates

Week – 10: ADDER/ SUBTRACTOR
Design a 4 – bit Adder / Subtractor

Week – 11: BINARY TO GRAY CONVERTER
Design and realization of a 4 – bit gray to Binary and Binary to Gray Converter.

Week – 12: TRUTH TABLES AND EXCITATION TABLES
Verification of truth tables and excitation tables.

Week – 13: SHIFT REGISTER
Design and realization of an 8 bit parallel load and serial out shift register using flip-flops

Week – 14: MULTIPLEXER
Design and realization of 8x1 using 2x1 MUX

IV. REFERENCE BOOKS:

1. Jacob Millman, Herbert Taub, Mothiki S PrakashRao, "Pulse Digital and Switching Waveforms", Tata McGraw-Hill, 3rd Edition, 2008.
2. David A. Bell, "Solid State Pulse Circuits", PHI, 4th Edition, 2002.
3. D Roy Chowdhury, "Linear Integrated Circuits", New Age International (p) Ltd, 2nd Edition, 2003.
4. Ramakanth A. Gayakwad, "Op-Amps & linear ICs", PHI, 3rd Edition, 2003.

V. WEB REFERENCES:

1. http://www.tedpavlic.com/teaching/osu/ece327/
2. http://www.ee.iitkgp.ac.in
3. http://www.citchennai.edu.in
4. http://american.cs.ucdavis.edu/academic/ecs154a.sum14/postscript/cosc205.pdf
5. http://www.ece.rutgers.edu/~marsic/Teaching/DLD/slides/lec-1.pdf

FUNDAMENTALS OF DATABASE SYSTEMS

IV Semester: CE / EEE / ME / ECE / AE

Course Code	Category	Hours / Week			Credits	Maximum Marks		
ACSC18	SKILL	L	T	P	C	CIA	SEE	Total
		-	-	-	-	-	-	-
Contact Classes: Nil	Tutorial Classes: Nil	Practical Classes: Nil				Total Classes: Nil		

I. COURSE OVERVIEW

The fundamentals of Database systems are vital components of modern information systems. Database applications all pervasive and range in size from small in-memory databases to terabytes or even larger in various applications domains. The course focuses and the fundamentals of knowledgebase and relational database management systems, and the current developments in database theory and their practices.

II. COURSE OBJECTIVES:

The course should enable the students to:

I. Understand the role of database management system in an organization and learn the database concepts.
II. Design databases using data modeling and data normalization techniques.
III. Construct database queries using relational algebra and calculus.
IV. Understand the concept of a database transaction and related database facilities.
V. Learn how to evaluate set of queries in query processing.

III. COURSE SYLLABUS:

MODULE: I CONCEPTUAL MODELING (10)

Introduction to file and database systems: Database system structure, data models: entity relationship model, relational model.

MODULE: II RELATIONAL APPROACH (08)

Relational algebra and calculus: Relational algebra, selection and projection, set operations, renaming, joins, division, examples of algebra queries, relational calculus, tuple relational calculus.

MODULE : III BASIC SQL QUERY AND NORMALIZATION (10)

SQL data definition; Queries in SQL: updates, views, integrity and security, relational database design.

Normal Forms: 1NF, 2NF, 3NF and BCNF.

MODULE : IV TRANSACTION MANAGEMENT (09)

Transaction processing: Introduction, need for concurrency control, desirable properties of transaction, schedule and recoverability, Serializability and schedules

MODULE : V CONCURRENCY CONTROL (08)

Concurrency control; Types of locks: Two phases locking, deadlock, timestamp based concurrency control, recovery techniques, concepts, immediate update, deferred update, shadow paging.

IV. TEXT BOOKS:

1. Abraham Silberschatz, Henry F. Korth, S. Sudarshan, "Database System Concepts", McGraw-Hill, 4thEdition, 2002.

V. REFERENCE BOOKS:

1. Ramez Elmasri, Shamkant B. Navathe, "Fundamental Database Systems", Pearson Education, 3rd Edition, 2003.
2. Raghu Ramakrishnan, "Database Management System", Tata McGraw-Hill Publishing Company, 3rd Edition, 2003.
3. Hector Garcia Molina, Jeffrey D. Ullman, Jennifer Widom, "Database System Implementation", Pearson Education, United States, 1st Edition, 2000.
4. Peter Rob, Corlos Coronel, "Database System, Design, Implementation and Management", Thompson Learning Course Technology, 5th Edition, 2003.

VI. WEB REFERENCES:

1. https://www.youtube.com/results?search_query=DBMS+onluine+classes
2. http://www.w3schools.in/dbms/
3. http://beginnersbook.com/2015/04/dbms-tutorial/

VII. E-TEXT BOOKS

1. http://www.e-booksdirectory.com/details.php?ebook=10166
2. http://www.e-booksdirectory.com/details.php?ebook=7400re

BUSINESS ECONOMICS AND FINANCIAL ANALYSIS

IV Semester: CSE / CSIT / CSE(DS), CSE(CS)
V Semester: AE / CE / EEE | VI Semester: ECE / ME / IT / CSE(AI&ML)

Course Code	Category	Hours / Week			Credits	Maximum Marks		
AHSC13	Foundation	L	T	P	C	CIA	SEE	Total
		3	-	-	3	30	70	100
Contact Classes: 45	Tutorial Classes: Nil	Practical Classes: Nil				Total Classes: 45		

Prerequisite: There is no prerequisite is required to this course

I. COURSE OVERVIEW:

The course is designed in such a way that it gives an overview of concepts of Economics. Managerial Economics enables students to understand micro environment in which markets operate how price determination is done under different kinds of competitions. Financial Analysis gives clear idea about concepts, conventions and accounting procedures along with introducing students to fundamentals of ratio analysis and interpretation of financial statements. Break Even Analysis is very helpful to the Business Concern for Decision Making, controlling and forward Strategic Planning. Ratio analysis gives an idea about financial forecasting, financial planning, controlling the business and decision making.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The concepts of business economics and demand analysist o helps in optimal decision making in business environment**.**
II. The functional relationship between Production and factors of production and able to compute breakeven point to illustrate the various uses of breakeven analysis.
III. The features, merits and demerits of different forms of business organizations existing in the modern business environment and market structures.
IV. The concept of capital budgeting and allocations of the resources through capital budgeting methods and compute simple problems for project management.
V. Various accounting concepts and different types of financial ratios for knowing financial positions of business concern.

III. COURSE OBJECTIVES:

MODULE – I: INTRODUCTION AND DEMAND ANALYSIS (07)

Definition, nature and scope of business economics; Demand analysis; Demand determinants, law of demand and its exceptions; Elasticity of demand: Definition, types, measurement and significance of elasticity of demand, demand forecasting, factors governing demand forecasting.

MODULE – II: PRODUCTION AND COST ANALYSIS (10)

Production function; Isoquants and isocosts, MRTS, least cost combination of inputs, Cobb-Dougles production function, internal and external economies of scale, cost analysis; Cost concepts: Break even analysis (BEA), determination of break-even point (simple problems), managerial significance.

MODULE – III: MARKETS AND NEW ECONOMIC ENVIRONMENT (08)

Types of competition and markets, features of perfect competition, monopoly and monopolistic competition, price-output determination in case of perfect competition and monopoly business.

Features and evaluation of different forms of business organizations: Sole proprietorship, partnership, joint stock company, public enterprises and their types.

MODULE – IV: CAPITAL BUDGETING (10)

Capital and its significance, types of capital, estimation of fixed and working capital requirements, methods and sources of raising capital, capital budgeting: features of capital budgeting proposals; Methods of capital budgeting: Payback period, accounting rate of return(ARR), net present value method and internal rate of return method (simple problems).

MODULE – V: INTRODUCTION TO FINANCIAL ACCOUNTING AND FINANCIAL ANALYSIS (10)

Financial accounting objectives, functions, importance; Accounting concepts and accounting conventions -double-entry book keeping, journal, ledger, trial balance; Final accounts: Trading account, profit and loss account and balance sheet with simple adjustments; Financial analysis: Analysis and interpretation of liquidity ratios, activity ratios, capital structure ratios and profitability ratios (simple problems), Du Pont chart.

IV. TEXT BOOKS:

1. Aryasri, "Managerial Economics and Financial Analysis", TMH publications, 4th Edition, 2012.
2. M. Kasi Reddy, Saraswathi, "Managerial Economics and Financial Analysis", PHI Publications, New Delhi, 2nd Edition, 2012.
3. Varshney, Maheswari, "Managerial Economics", Sultan Chand Publications, 11th Edition, 2009.

V. REFERENCE BOOKS:

1. S. A. Siddiqual, A. S. Siddiqual, "Managerial Economics and Financial Analysis", New Age International Publishers, Hyderabad, Revised 1st Edition, 2013.
2. S. N. Maheswari, S. K. Maheswari, "Financial Accounting", Vikas publications, 3rd Edition, 2012.
3. J. V. Prabhakar Rao, P. V. Rao, "Managerial Economics and Financial Analysis", Maruthi Publishers, Reprinted Edition, 2011.
4. Vijay Kumar, Appa Rao, "Managerial Economics and Financial Analysis", Cengage Publications, 1st Edition, Paperback, 2011.

VI. WEB REFERENCES:

1. https:// www.slideshare.net/glory1988/managerial-economics-and- financial analysis
2. https:// thenthata.web4kurd.net/mypdf/managerial-economics-and- financial analysis
3. https:// bookshallcold.link/pdfread/managerial-economics-and-financial analysis
4. https:// www.gvpce.ac.in/syllabi/Managerial Economics and financial analysis

ELECTRICAL POWER TRANSMISSION SYSTEMS

<table>
<tr><td colspan="10">V Semester: EEE</td></tr>
<tr><th>Course Code</th><th>Category</th><th colspan="3">Hours / Week</th><th>Credits</th><th colspan="3">Maximum Marks</th></tr>
<tr><td rowspan="2">AEEC15</td><td rowspan="2">Core</td><td>L</td><td>T</td><td>P</td><td>C</td><td>CIA</td><td>SEE</td><td>Total</td></tr>
<tr><td>3</td><td>1</td><td>-</td><td>4</td><td>30</td><td>70</td><td>100</td></tr>
<tr><td>Contact Classes: 45</td><td>Tutorial Classes: 15</td><td colspan="4">Practical Classes: Nil</td><td colspan="3">Total Classes: 60</td></tr>
<tr><td colspan="9">Prerequisites: Network Analysis (AEEC05), Electrical Power Generation Systems (AEEC10)</td></tr>
</table>

I. COURSEOVERVIEW:

Electrical Power Transmission Systems deals with the modeling, analysis and design of electrical power transmission lines. It gives an emphasis on overhead line insulators, underground cables, transient behavior ofthe lines, corona phenomena, Extra High Voltage Alternating Current (EHVAC) and High Voltage Direct Current (HVDC) transmission systems.

II. COURSEOBJECTIVES:

The students will try to learn:

I The mathematical solutions for transmission line parameters of a single phase and three phase system.

II The mechanical design of overhead transmission lines, the use of insulators and underground cables inelectrical power transmission system.

III The mathematical modeling of short, medium and long transmission lines along with the transient behavior.

IV The Extra High Voltage Alternating Current (EHVAC) and High Voltage Direct Current (HVDC) transmission systems used for transmitting electrical power to consumers.

III. COURSESYLLABUS:

MODULE-I: TRANSMISSION LINE PARAMETERS (08)

Transmission line parameters: Types of conductors, calculation of resistance for solid conductors, description and effect of resistance on solid conductors, calculation of inductance for single phase and three phase, single and double circuit lines, concept of GMR, GMD, symmetrical and asymmetrical conductor configuration with and without transposition, Skin and Proximity effect ; Numerical Problems: Calculation of capacitance for 2 wire and 3 wire systems, effect of ground on capacitance, capacitance calculations for symmetrical and asymmetrical single and three phase, single and double circuit lines, numerical problems.

MODULE-II: MECHNAICAL DESIGN OF TRANSMISSION LINES (10)

Overhead line insulators: Types of insulators, string efficiency and methods for improvement, numerical problems, voltage distribution, calculation of string efficiency, capacitance grading and static shielding, testing of insulators; Sag and tension calculations: Sag and tension calculations with equal and unequal heights of towers, effect of wind and ICE on weight of conductor, numerical problems, stringing chart and sag template and its applications; mechanical design of typical towers and conductors for 400KV, 220KV and 132KV operations.

MODULE-III: PERFORMANCE OF TRANSMISSION LINES (09)

Performance of short and medium length transmission lines: Classification of transmission lines, short, medium and long line and their model representations, nominal-T, nominal-Pie and A, B, C, D constants for symmetrical and asymmetrical networks, numerical problems, mathematical solutions to estimate regulation and efficiency of all types of lines, numerical problems.

Performance of long transmission lines: Long transmission line, rigorous solution, evaluation of A, B, C, D constants, representation of long lines, equivalent-T and equivalent Pie network models (numerical problems); Ferranti effect, charging current, effect on regulation of the transmission line, surge impedance and SIL of long lines, wave length and velocity of propagation of waves.

MODULE-IV: POWER SYSTEM TRANSIENTS AND FACTORS GOVERNING PERFORMANCE OF TRANSMISSION LINES (10)

Power systems transients: Incident reflected and refracted waves, Types of system transients, travelling or propagation of surges, attenuation, distortion, reflection and refraction coefficients, termination of lines with different types of conditions, open circuited line, short circuited line, T-junction, lumped reactive junctions (numerical problems), Bewley"s lattice diagrams (for all the cases mentioned with numerical examples); Corona, description of the phenomenon, factors affecting corona, critical voltages and power loss, radio interference, Electrostatic and electromagnetic interference with communication lines.

MODULE-V: UNDERGROUND CBALES, EHV TRANSMISSION AND HVDC TRANSMISSION (08)

Underground cables: Types of cables, construction, types of insulating materials, calculation of insulation resistance and stress in insulation, numerical problems, capacitance of single and 3core belted cables, numerical problems, grading of cables, capacitance grading, numerical problems, description of inter-sheath grading, HV cables. Need of EHV transmission systems, types of DC links, comparison of AC and DC transmission, advantage of DC transmission, HVDC systems in India.

IV. TEXTBOOKS:

1. D P Kothari and I J Nagrath, "Power System Engineering", McGraw-Hill Education, 2ndEdition, 2007.
2. V K Mehta and Rohit Mehta, "Principles of Power System", S Chand, 3rd revised Edition, 2015.
3. D Das, "Electrical Power systems", New age international publishers, 2nd Edition, 2006.
4. K R Padiyar, "HVDC transmission Systems", New age international publishers, 2nd Edition, 2005.
5. B R Guptha, "Power system analysis and Design" S. Chand Publishing, 2nd Edition, 1998.

V. REFERENCEBOOKS:

1. C L Wadhwa, "Electric Power Systems", New age publications, New Delhi, 9thEdition, 2007.
2. TuranGonen, "Electrical Power Distribution System Engineering", CRC Press, 3rdEdition, 2014.
3. V Kamaraju, "Electrical Power Distribution Systems", TMH, Publication, Edition 2009.
4. Singh S N, "Electric Power Generation, Transmission and Distribution", Prentice Hall of India Pvt. Ltd., New Delhi, 2ndEdition, 2002**.**

VI. WEBREFERENCES:

1. https://www.en.wikipedia.org/wiki/Electric_power_transmission
2. https://www.iec.ch/about/brochures/pdf/technology/transmission.pdf
3. https://www.teriin.org/upfiles/pub/papers/ft33.pdf

VII. E-TEXTBOOKS:

1. https://www.jfgieras.com/Grigsby_Chapter_34_LEM.pdf
2. https://www.personal.psu.edu/sab51/vls/vonmeier.pdf
3. https://www.edsonjosen.dominiotemporario.com/doc/Livro_Electric_Power_Distribution_System_En gineering_-_Turan_Gonen.pdf

POWER ELECTRONICS

<table>
<tr><td colspan="9">V Semester: EEE</td></tr>
<tr><th>Course Code</th><th>Category</th><th colspan="3">Hours / Week</th><th>Credits</th><th colspan="3">Maximum Marks</th></tr>
<tr><td rowspan="2">AEEC16</td><td rowspan="2">Core</td><td>L</td><td>T</td><td>P</td><td>C</td><td>CIA</td><td>SEE</td><td>Total</td></tr>
<tr><td>3</td><td>1</td><td>-</td><td>4</td><td>30</td><td>70</td><td>100</td></tr>
<tr><td>Contact Classes: 45</td><td>Tutorial Classes: 15</td><td colspan="4">Practical Classes: Nil</td><td colspan="3">Total Classes: 60</td></tr>
<tr><td colspan="9">Prerequisites: Network Analysis, Digital Electronics</td></tr>
</table>

I. COURSE OVERVIEW:

The course focuses on presenting concepts for conversion, control and monitoring of electrical energyusing power semiconductor devices. Methods for analyzing power electronic converters suitable for DC/AC, DC/DC, AC/AC and AC/DC electrical energy conversions including regulators are presented. Additionally, principles for designing power electronic converters, including their power semiconductors and passive elements are established. The applications of power electronics in the fields of sustainable energy technologies, switched mode power supplies and uninterruptible power supplies as well as application of power electronic converters for transmission, distribution and control in the power systems is described.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The concepts on power semiconductor devices related to its characteristics, ratings, and protection to select these devices for various applications.
II. The fundamental principles and control techniques of power electronic converters for analyzing AC/DC, DC/DC, AC/AC and DC/AC power conversion circuits.
III. The application of power electronic converters in the fields of battery management system, industrial drive applications and enhancement of power quality.

III. COURSE SYLLABUS

MODULE-I: POWER SWITCHING DEVICES (09)

Thyristor, MOSFET, IGBT: I-V Characteristics; R, RC and UJT firing circuit for thyristor; Gate drive circuits for MOSFET and IGBT. Series and parallel operation of thyristors, ratings, protection against dv/dt and di/dt, design of Snubber circuit, forced commutation circuits, other devices in thyristor family: TRIAC, GTO and their characteristics, numerical problems.

MODULE –II: PHASE CONTROLLED RECTIFIERS (09)

Single phase half wave and single phase full wave (Mid-Point and Bridge configurations) thyristor rectifier with R- load and highly inductive load; derivation of average load voltage and current, effect of freewheeling diode, effect of source inductance

MODULE-III: CHOPPERS (09)

Basic chopper operation, control strategies, step up chopper, derivation of load voltage and load currents with R and RL loads, chopper configurations.

Power circuit of a buck, boost and buck-boost converters: Analysis and waveforms at steady state.

MODULE –IV: AC VOLTAGE CONTROLLER AND CYCLO CONVERTERS (09)

Single phase AC voltage controllers - two SCRs in anti-parallel with R and RL loads, derivation of rms load voltage and load current, numerical problems, Cyclo converters - single phase midpoint and bridge type (step-up and step-down operations) with R and RL loads.

MODULE –V: INVERTERS (09)

Single phase inverters: Basic operation, voltage source inverters, basic series and parallel inverters, current source inverter, modified Mc Murray and Mc Murray-Bedford half bridge inverters (operation and waveforms), voltage control by pulse width modulation techniques (single pulse, multiple pulse and sinusoidal), numerical problems. Three phase bridge Inverters - 180^o and 120^o conduction modes of operation.

IV. TEXT BOOKS:

1. Dr. P S Bimbhra, "Power Electronics", Khanna Publishers, Delhi, 4thEdition, 2008.
2. M H Rashid, "Power electronics: circuits, devices, and applications", Pearson Education India, 3rdEdition, 2009.

V. REFERENCE BOOKS:

1. L Umanand, "Power Electronics: Essentials and Applications", Wiley India, 3rd Edition, 2009.
2. N Mohan and T M Undeland, "Power Electronics: Converters, Applications and Design", John Wiley & Sons, 2nd Edition, 2007.
3. R. W. Erickson and D Maksimovic, "Fundamentals of Power Electronics", Springer Science & Business Media, 2nd Edition, 2007.

V. WEB REFERENCES:

1. https://www.coursera.org/learn/power-electronics
2. https://nptel.ac.in/courses/108/102/108102145/
3. https://www.electronicsforu.com/videos-slideshows/power-electronic-devices
4. https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-334-power-electronics-
5. spring-2007/lecture-notes/

MICROPROCESSORS AND MICROCONTROLLERS

<table>
<tr><td colspan="10">V Semester: EEE</td></tr>
<tr><th rowspan="1">Course Code</th><th>Category</th><th colspan="3">Hours / Week</th><th>Credits</th><th colspan="3">Maximum Marks</th></tr>
<tr><td rowspan="2">AECC19</td><td rowspan="2">Core</td><td>L</td><td>T</td><td>P</td><td>C</td><td>CIA</td><td>SEE</td><td>Total</td></tr>
<tr><td>3</td><td>1</td><td>-</td><td>4</td><td>30</td><td>70</td><td>100</td></tr>
<tr><td>Contact Classes: 45</td><td>Tutorial Classes: 15</td><td colspan="4">Practical Classes: Nil</td><td colspan="3">Total Classes: 60</td></tr>
<tr><td colspan="10">Prerequisites: Analog Electronics (AECC07), Digital Electronics (AECC16)</td></tr>
</table>

I. COURSE OVERVIEW:

To develop an in-depth understanding of the structure and operations of microprocessor and microcontrollers, machine language programming and interfacing techniques. Through understanding of hardware and software interaction and integration.

II. COURSE OBJECTIVES:

The students will try to learn:

I. Imbibe sound knowledge about architecture, instruction set and concepts of 8086 and 8051.
II. Demonstrate the ability to develop programs for different applications using assembly language of 8086 and 8051.
III. Impart knowledge of different types of external peripherals like 8255, 8259, 8279, 8251, 8257.
IV. Proficient in Memory and I/O interfacing with 8086 and 8051.

III. COURSE SYLLABUS

MODULE-I: 8086 MICROPROCESSORS (09)

Register organization of 8086, Architecture, signal description of 8086, physical memory organization, general bus operation, I/O addressing capability, special purpose activities, Minimum mode, maximum mode of 8086 system and timings, machine language instruction formats, addressing mode of 8086, instruction set off 8086, assembler directives and operators.

MODULE –II: PROGRAMMING WITH 8086 MICROPROCESSOR (09)

Machine level programs, programming with an assembler, Assembly language programs, introduction to stack, stack structure of 8086/8088, interrupts and interrupt service routines. Interrupt cycle of 8086, non-mask able interrupt and mask able interrupts, interrupt programming

MODULE-III: INTERFACING WITH 8086/88 (08)

Semiconductor memory interfacing, dynamic RAM interfacing, interfacing i/o ports, PIO 8255 modes of operation of 8255, interfacing to D/A and A/D converters, stepper motor interfacing, control of high-power devices using 8255.

Programmable interrupt controller 8259A, the keyboard /display controller8279, programmable communication interface 8251 USART, DMA Controller 8257.

MODULE –IV: 8051 MICROCONTROLLER (10)

8051 Microcontroller – Internal architecture and pin configuration, 8051 addressing modes, instruction set, Bit addressable features. I/O Port structures, assembly language programming using data transfer, arithmetic, logical and branch instructions.

MODULE –V: SYSTEM DESIGN USING MICROCONTROLLER (10)

8051 Timers/Counters, Serial data communication and its programming, 8051 interrupts, Interrupt vector table, Interrupt programming. Real world interfacing of 8051 with external memory, expansion of I/O ports, LCD, ADC, DAC, stepper motor interfacing

IV. TEXT BOOKS:

1. Ray A.K, Bhurchandi K.M, “Advanced Microprocessor and Peripherals”, TMH, 2nd Edition, 2012
2. Muhammad Ali Mazidi, J.G. Mazidi, R.D McKinlay,” The 8051 Microcontroller and Embedded systems using Assembly and C”, Pearson education, 2nd Edition, 2009.
3. Douglas V. Hall, “Microprocessors and Interfacing Programming and Hardware”, TMGH, 2nd Edition, 1994.

V. REFERENCE BOOKS:

1. Kenneth J. Ayala, “The 8051 Microcontroller”, Thomson Learning, 3rd Edition, 2005.
2. Manish K. Patel, “The 8051 Microcontroller Based Embedded Systems”, McGraw Hill, 1st Edition, 2014.
3. Ajay V Deshmukh, “Microcontrollers”, TATA McGraw Hill publications, 2nd Edition, 2012.

V. WEB REFERENCES:

1. http://www.nptel.ac.in/downloads/106108100/
2. http://www.the8051microcontroller.com/web-references
3. http://www.iare.ac.in

ELECTRICAL MACHINE DESIGN

V Semester: EEE								
Course Code	Category	Hours /Week			Credits	Maximum Marks		
AEEC17	Elective	L	T	P	C	CIA	SEE	Total
		3	-	-	3	30	70	100
Contact Classes:45	Tutorial Classes:Nil	Practical Classes: NIL				Total Classes: 45		
Prerequisite: DC machines and Transformers (AEEC07), AC Machines (AEEC11)								

I. COURSE OVERVIEW

This course deals with basic design principles of electrical machines including DC machines, Transformers, Induction machines and Synchronous machines. It also covers the design of electrical machine parameters such as main dimensions, magnetic, electric loading, number of slots and winding dimensions. Thermal and magnetic circuit design of electric machines will be also covered. The students will be familiarized with the computer aided design of electrical machines and FEM software

II. OBJECTIVES:

The students will try to learn:

I. The electrical engineering materials and thermal considerations of materials.
II. The design of core, yoke, windings and cooling systems of AC-DC machines and transformers.
III. The use of softwaretools like finite element and computer aided design software for machine design.

III. COURSE SYLLABUS:

MODULE-I: INTRODUCTION (09)

Major considerations in electrical machine design, electrical engineering materials, space factor, choice of specificelectricaland magneticloadings, thermal considerations, heatflow,temperaturerise,ratingof machines.

MODULE-II: DESIGN OF TRANSFORMERS (09)

Sizing of a transformer, main dimensions, KVA output for single and three-phase transformers, window spacefactor, overall dimensions, operating characteristics, regulation, no load current, temperature rise in transformers,designofcoolingtank, methods for coolingof transformers.

MODULE-III: DESIGN OF INDUCTION MOTOR (09)

Sizing of an induction motor, dimensions, length of air gap, rules for selecting rotor slots of squirrel cage machines, design of rotor bars &slots, design of end rings.

Design of wound rotor, leakage reactance of poly phase machines, magnetizing current, short circuit current, circle diagram, operating characteristics.

MODULE-IV: DESIGN OF INDUCTION MOTOR (09)

Sizing of a synchronous machine, main dimensions, design of salient pole machines, short circuit ratio, shape ofpole face, armature design, armature parameters, estimation of air gap length, design of rotor, design of damperwinding,determinationoffullloadfieldMMF,designoffieldwindingdesignofturboalternators,rotordesign.

MODULE-V: COMPUTER AIDED DESIGN (09)

Limitations (assumptions) of traditional designs, need for CAD analysis, synthesis and hybrid methods, designoptimization methods, variables, constraints and objective function, problem formulation. Introduction to FEMbased machine design. Introduction to complex structures of modern machines-PMSMs, BLDCs, SRM and claw-polemachines.

IV. TEXT BOOKS:

1. AK Sawhney, "A Course in Electrical Machine Design", Dhanpat Rai and Sons, 1st Edition, 1970.
2. MGSay, "Theory & Performance & Design of A.C. Machines", ELBS London, 2nd Edition, 1998.
3. K M V Murthy, "Computer Aided Design of Electrical Machines", BS Publications, 2nd Edition, 2008.

V. REFERENCEBOOKS:

1. A Shanmuga Sundaram, G Gangadharan and R Palani, "Electrical Machine Design Data Book", New Age International, 1st Edition, 1979.
2. SK Sen, "Principles of Electrical Machine Design with Computer Programmes", Oxford and IBH Publishing, 2ndEdition, 2006.
3. KLNarang, "AText Book of Electrical Engineering Drawings", Satya Prakashan, 1st Edition, 1969.

VI. WEB REFERENCES

1. https://www.electrical4u.com
2. https://www.iare.ac.in
3. https://www.researchgate.net
4. https://www.cusp.umn.edu

VII. E-TEXT BOOKS:

1. https://www.jntubook.com/
2. https://www.freeengineeringbooks.com

COMPUTATIONAL ELECTROMAGNETICS

V Semester: EEE								
Course Code	Category	Hours / Week			Credits	Maximum Marks		
AEEC18	Elective	L	T	P	C	CIA	SEE	Total
		3	0	0	3	30	70	100
Contact Classes: 45	Tutorial Classes: Nil	Practical Classes: Nil				Total Classes: 45		
Prerequisite: Electromagnetic Fields								

I. COURSE OVERVIEW:

This course covers the mathematical formulation of Integral Equations Methods (and their solution by the Method of Moments), the Finite Element Method, and the Finite Difference Time Domain method. These methods are illustrated by their use in solving scattering problems and antenna radiation/impedance calculation problems. Additional topics include introduction to inverse problems, calculating the mutual coupling between antennas, finding the electromagnetic modes of a waveguide, and techniques to hybridize the Finite Element Method with the Integral Equation Method.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The scope of contemporary and emerging application areas in electromagnetic wave technology, especially high-speed electronic and optical communications.
II. The basic computational algorithm in electromagnetics with finite element method.
III. The computational skills in applied electromagnetics and related disciplines and ability not only to effectively use electromagnetic software, but also to understand the foundations of various codes.
IV. The conceptual approaches for numerical dispersion and stability of Finite difference time domine electromagnetic wave simulations.
V. The real-world applications of modern computational tools in electromagnetic scattering, propagation, and radiation.

III. COURSE SYLLABUS:

MODULE-I: BASIC ELECTROMAGNETICS PRINCIPLES (8)

Maxwells equations, applications of Computational Electromagnetics, electrostatics and Magnetostatics, wave equation and propagation, scalar and vector potentials, surface equivalence principle, greens Function, boundary conditions, linear algebra for computational EM.

MODULE-II: INTEGRAL EQUATION METHODS (10)

Introduction to integral equations, surface integral equations: mathematical derivation of Huygen's principle, extinction theorem, introduction to Green's functions: 1D example of string, 2D and 3D wave equation, solving integral equations using the method of moments (MoM), summary of integral equation methods; computation of radar cross-section (RCS).

MODULE-III: FINITE ELEMENT METHODS (FEM) (9)

Introduction and history of FEM, FEM in the method of moments framework; 1 and 2D basis functions in FEM, weak form of FEM; Robin boundary conditions; example of solving the 1D wave equation using FEM, 2D edge-based (vector) FEM.

Weak form of FEM; shape functions, weak form and radiation boundary conditions, total and scattered field formulations, assembly of equations, numerical aspects in computing 2D FEM matrix elements, and overall procedure.

MODULE-IV: FINITE DIFFERENCE TIME DOMAIN (FDTD) METHOD (10)

Basics of FDTD: Finite Differences, FDTD solution of Maxwell Equations in simple cases, scattered field vs total field formulation, implementation issues, outer radiation boundary conditions, Yee cell, Yee algorithm for 3D formulation of FDTD, Perfectly Matched Layer (PMC) ABC, frequency domain results from time domain output of FDTD, computational issues: Stair casing errors, numerical dispersion, frequency dependent materials.

MODULE-V: APPLICATIONS OF COMPUTATIONAL ELECTROMAGNETICS (8)

Microwave inverse imaging, antenna radiation problems: Hertz dipole and its fields, Pocklington's integral equation for finding the current on a finite length wire, mutual coupling between array elements. Calculating the modes of a waveguide structure using the integral equation method: solving a generalized eigen value problem. Hybrid methods in CEM: Finite Element - Boundary Integral method

IV. TEXT BOOKS

1. A. Taflove and SC Hagness, "Computational Electrodynamics: The Finite Difference Time Domain Method", Artech House, 3rd Edition, 2007.
2. Andrew F. Peterson, Scott L. Ray, Raj Mittra, "Computational Methods for Electromagnetics", IEEE Press Series on Electromagnetic Wave Theory, 1st Edition, 2013.
3. Walton C. Gibson: "The Method of Moments in Electromagnetics", Chapman and Hall, 1st Edition, 2013.

V. REFERENCEBOOKS:

1. Andrew F. Peterson, Scott L. Ray, Raj Mittra, "Computational Methods for Electromagnetics", Universities Press, 1st Edition, 2001.
2. Stephen D. Gedney "Introduction to the Finite-Difference Time-Domain (FDTD) Method for Electromagnetics" Morgan & Claypool, 2011
3. Karl F. Warnick: "Numerical Methods for Engineering: An Introduction Using MATLAB and Computational Electromagnetics Examples," SciTech Publishing, 2010.
4. J. M. Jin, "The Finite Element Method in Electromagnetics", John Wiley & Sons, New York, 2nd Edition., 2002.

VI. WEBREFERENCES:

1. https://www.researchgate.net
2. https://www.aar.faculty.asu.edu/classes
3. https://www.facstaff.bucknell.edu/
4. https://www.electrical4u.com
5. https://www.iare.ac.in

VII. E-TEXTBOOKS:

1. https://www.springer.com
2. https://www.ieeexplore.ieee.org

SPECIAL ELECTRICAL MACHINES

V Semester: EEE								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
AEEC19	**Elective**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		3	-	-	3	30	70	100
Contact Classes: 45	**Tutorial Classes: Nil**	**Practical Classes: NIL**				**Total Classes: 45**		
Prerequisite: DC machines and Transformers AC Machines								

I. COURSE OVERVIEW

This course deals with basic design principles of electrical machines including synchronous reluctance motor, switched reluctance, permanent magnet synchronous motors and permanent magnet brushless DC motors. It also facilitates the study of special machines which are the major part of industrial applications

II. OBJECTIVES:

The students will try to learn:

I. The construction, principle of operation and performance of special machines.
II. The performance characteristics of power converters for switched reluctance motors.
III. The power converters and their controllers for permanent magnet brushless DC motors.

III. COURSE SYLLABUS:

MODULE-I: SYNCHRONOUS RELUCTANCE MOTORS (09)

Reluctance motors: Constructional features, types, axial and radial flux motors, operating principles, variable reluctance motors, voltage and torque equations, phasor diagram, performance characteristics, applications

MODULE-II: STEPPER MOTOR (09)

Stepper motors: Constructional features, principle of operation, variable reluctance motor, hybrid motor, single and multi-stack configurations, torque equations, modes of excitation, characteristics, drive circuits, microprocessor control of stepper motors, closed loop control, concept of lead angle, applications.

MODULE-III: SWITCHED RELUCTANCE MOTORS (09)

Switched reluctance motors: Constructional features: rotary and linear SRM, principle of operation, torque production, steady state performance prediction, analytical method, power converters and their controllers.

Methods of rotor position sensing: sensor less operation, characteristics and closed loop control, applications.

MODULE-IV: PERMANENT MAGNET BRUSHLESS D.C. MOTORS (09)

BLDC motors: Permanent magnet materials, minor hysteresis loop and recoil line, magnetic Characteristics, permeance, coefficient, principle of operation, types, magnetic circuit analysis, EMF and torque equations, commutation, power converter circuits and their controllers, motor characteristics and control, applications.

MODULE-V: PERMANENT MAGNET SYNCHRONOUS MOTORS (09)

Permanent magnet synchronous motors: Principle of operation, ideal PMSM, EMF and torque equations, armature MMF, synchronous reactance, sine wave motor with practical windings, phasor diagram, torque speed characteristics, power controllers, converter volt ampere requirements, applications

IV. TEXT BOOKS:

I. T.J.E. Miller, “Brushless Permanent Magnet and Reluctance Motor Drives”, Clarendon Press, Oxford, 1989.
II. T. Kenjo, “Stepping Motors and Their Microprocessor Controls”, Clarendon Press London, 1984.

V. REFERENCEBOOKS:

1. R. Krishnan, “Switched Reluctance Motor Drives Modeling, Simulation, Analysis, Design and Application”, CRC Press, NewYork, 2001.
2. P.P. Aearnley, “Stepping Motors –A Guide to Motor Theory and Practice”, Peter Perengrinus London, 1982.

3. T. Kenjo and S. Nagamori, “Permanent Magnet and Brushless DC Motors”, Clarendon Press, London, 1988

VI. WEB REFERENCES

1. https://www.electrical4u.com
2. https://www.iare.ac.in
3. https://www.researchgate.net
4. https://www.cusp.umn.edu

VII. E-TEXT BOOKS:

1. https://www.jntubook.com/
2. https://www.freeengineeringbooks.com

ELECTRICAL ENERGY CONSERVATION AND AUDITING

V Semester: EEE

Course Code	Category	Hours / Week			Credits	Maximum Marks		
AEEC20	Elective	L	T	P	C	CIA	SEE	Total
		3	-	-	3	30	70	100
Contact Classes: 45	Tutorial Classes: Nil	Practical Classes: Nil				Total Classes: 45		

Prerequisite: Fundamentals of Courses like Power Systems, Electrical Machines, etc

I. COURSE OVERVIEW:

The course provides basic understanding of energy audit and management. The consumption of energy is increasing day by day. One way to cope up with the increase in energy demand is to increase the production of energy which demands more investment and the other way is to conserve the energy as energy conserved/saved is twice the energy generated. Energy conservation means reduction in energy consumption but not compromising with the quality or quantity of energy production. Essential theoretical and practical knowledge about the concept of energy conservation, energy management, different approaches of energy conservation in industries, economic aspects of energy conservation project and energy audit and measuring instruments in commercial and industrial sector will be achieved through this course.

II. COURSE OBJECTIVES:

The students will try to learn:

I. A clear conceptual understanding of technical and commercial aspects of energy conservation and energy auditing.
II. The Managerial skills to assess feasibility of alternative approaches and drive strategies regarding energy conservation and energy auditing.
III. The energy management skills and strategies in the energy management system.
IV. The Various energy conservation methods useful in a particular industry.
V. Appropriate energy conservation method for the critical area identified and able to prepare an energy audit report.

III. COURSE SYLLABUS:

MODULE-I: ENERGY SCENARIO (09)

Commercial and Non-commercial energy: Primary energy resources, energy needs of growing economy, long, medium- and short-term energy scenarios, energy pricing, energy sector reforms, energy and environment, energy security, conservation and its importance, restructuring of the energy supply sector, energy strategy for the future, air pollution, climate change.

MODULE -II: ENERGY MANAGEMENT AND AUDIT (12)

Energy audit: Need, types, approach understanding energy costs, bench marking, maximizing system efficiencies, optimizing the input energy requirements, fuel &energy substitution, energy audit instruments. Material and Energy balance: Facility as an energy system, methods for preparing process flow, material and energy balance diagrams.

MODULE -III: ENERGY EFFICIENCY IN ELECTRICAL SYSTEMS (08)

Electrical system: Electricity billing, electrical load management and maximum demand control, power factor improvement and its benefit, distribution and transformer losses.

Electric motors: Types, efficiency, factors of performance, rewinding and motor replacement issues.

MODULE -IV: ENERGY EFFICIENCY IN INDUSTRIAL SYSTEMS (08)

Compressed air system, Types of air compressors, efficiency, compressed air system components, energy saving opportunities in HVAC; Fans and blowers: Types, efficient system operation and energy conservation opportunities; Cooling tower: Types, efficient system operation, flow control strategies and energy saving opportunities.

MODULE -V: ENERGY EFFICIENT TECHNOLOGIES IN ELECTRICAL SYSTEMS (08)

Maximum demand controllers, Automatic power factor controllers, energy efficient motors, soft starters with energy saver, variable speed drives, energy efficient transformers, electronic ballast, energy efficient lighting controls.

IV. TEXT BOOKS:

1. Anthony J Pansini, Kenneth D Smalling, "Guide to Electric Load Management", Pennwell Publication, 2nd Edition, 1998.
2. E Jordan, "Energy-Efficient Electric Motors and Their Applications", Plenum Publication, corp, 2nd Edition, 1994.

V. REFERENCE BOOKS:

1. Y P Abbi, and Shashank Jain, "Energy Audit and Environment Management", Hand book on, 2nd Edition 2006.
2. S C Tripathy, "Utilization of Electrical Energy and Conservation", McGraw Hill, 1st Edition, 1991.
3. Albert Thumann, William J Younger, Terry Niehus, "Hand book of Energy Audits", 2nd Edition, 2009.
4. Giovanni Petrecca, "Industrial Energy Management", "Principles and Applications", The Kluwer international series - 207, 2nd Edition, 1999.

VI. WEB REFERENCES:

1. http://www.rroij.com/open-access/energy-conservation-andaudita-case-study.php?aid=42307
2. http://www.ijsrp.org/research-paper-0813/ijsrp-p2044.pdf
3. https://beeindia.gov.in/sites/default/files/1Ch3.pdf
4. https://www.slideshare.net/rayvarun/energy-conservation-ppt-by-vp-singh
5. http://elion.co.in/elion-energy-audit-conservation

VII. E-TEXT BOOKS:

1. https://www.amazon.in/ENergy-conservation-audit-b-patil-ebook/ dp/B07 hmvx5yv
2. https://www.worldcat.org/title/energy-management-audit-and-conservation/oclc/891484955

EXPERIENTIAL ENGINEERING EDUCATION (ExEEd) – PROJECT BASED LEARNING

<table>
<tr><td colspan="9">V Semester: Common for all branches</td></tr>
<tr><th>Course Code</th><th>Category</th><th colspan="3">Hours / Week</th><th>Credits</th><th colspan="3">Maximum Marks</th></tr>
<tr><td rowspan="2">ACSC20</td><td rowspan="2">Foundation</td><td>L</td><td>T</td><td>P</td><td>C</td><td>CIA</td><td>SEE</td><td>Total</td></tr>
<tr><td>2</td><td>-</td><td>-</td><td>1</td><td>30</td><td>70</td><td>100</td></tr>
<tr><td>Contact Classes: 36</td><td>Tutorial Classes: Nil</td><td colspan="4">Practical Classes: Nil</td><td colspan="3">Total Classes: 36</td></tr>
<tr><td colspan="9">Prerequisite: There are no prerequisites to take this course</td></tr>
</table>

I. COURSE OVERVIEW:

Project-based learning (PBL) is collaborative, learner-centered instructional approach where students work in groups to construct their knowledge using modern tools. It often requires students to collaborate, design, revise, and share their ideas and experiences with authentic audiences and supportive peer groups rather than collect resources, organize work, and manage long-term activities. Project-Based Learning begins with the assignment of tasks that will lead to the problem identification, modeling, simulation and analyzing the results.

II. COURSE OBJECTIVES:

The students will try to learn:

I. To emphasize learning activities that is long-term, interdisciplinary and student-centric.
II. To inculcate independent learning by problem solving with social context.
III. To engages students in rich and authentic learning experiences.
IV. To provide every student the opportunity to get involved either individually or as a group so as to develop team skills and learn professionalism.

III. COURSE SYLLABUS

I. Defining the Problem
II. Gathering requirements
III. Design / *Mode*ling
IV. Implementation
V. Testing
VI. Report

POWER ELECTRONICS LABORATORY

V Semester: EEE								
Course Code	Category	Hours / Week			Credits	Maximum Marks		
AEEC21	Core	L	T	P	C	CIA	SEE	Total
		-	-	3	1.5	30	70	100
Contact Classes: Nil	Tutorial Classes: Nil	Practical Classes: 36				Total Classes: 36		
Prerequisites: Power Electronics								

I. COURSE OVERVIEW:

This course is intended for practical experience by conduction experiments on rectifiers, inverters, choppers, AC voltage controllers and cycloconverters. It provides hands-on experience by examining the electrical characteristics of various power converters. The power electronic converter applications have been analyzed with simulation tools.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The engineering skills by way of electrical circuit design with power electronic devices and components.
II. Simulation and testing the different power converter circuits using simulation tools.
III. The demonstration of basic power electronic circuits for developing complex power converter modules.

III. COURSE SYLLABUS:

Expt. 1: SCR, MOSFET AND IGBT
Study the characteristics of SCR, MOSFET and IGBT.

Expt. 2: GATE FIRING CIRCUITS
Study the operation of gate firing circuits of SCR.

Expt. 3: HALF CONTROLLED CONVERTER
Study the performance characteristics of single-phasehalf-controlled converter with R and RL loads.

Expt. 4: FORCED COMMUTATION CIRCUITS
Plot the characteristics of forced commutation circuits (Class A, Class B, Class C, Class D and Class E).

Expt. 5: FULLY CONTROLLED BRIDGE CONVERTER
Study the characteristics of single phase fully controlled bridge converter with R and RL loads.

Expt. 6: SERIES INVERTER
Study the characteristics of single-phase series inverter with different loads.

Expt. 7: PARALLEL INVERTER
Study the characteristics of single-phase parallel inverter with different loads.

Expt. 8: AC VOLTAGE CONTROLLER
Plot the characteristics of Single-phase AC voltage controller with R and RL loads.

Expt. 9: DUAL CONVERTER
Study the characteristics of single-phase dual converter with R and RL loads.

Expt. 10: CYCLOCONVERTER
Study the characteristics of single phase cycloconverter with R and RL loads.

Expt. 11: THREE PHASE CONVERTERS
Plot the characteristics of three phase half converter with R and RL loads.

Expt. 12: MOSFET BASED CHOPPERS
Study the principle of operation of step-down chopper using MOSFET.

Expt. 13: SIMULATION OF THREE PHASE FULL CONVERTER AND PWM INVERTER
Simulation of three phase full converter and PWM inverter with R and RL loads by using MATLAB.

Expt. 14: SIMULATION OF BUCK – BOOST CHOPPER
Simulation of boost, buck, buck boost converter with R and RL loads by using MATLAB.

IV. REFERENCE BOOKS:

1. M D Singh, K B Kanchandhani, "Power Electronics", Tata McGraw Hill Publishing Company, 2nd Edition, 1998.
2. Dr. P S Bimbhra, "Power Electronics", Khanna Publishers, 5th Edition, 2012.
3. Ned Mohan, Tore M Undeland, William P Robbins, "Power Electronics: Converters, Applications and Design", 3rd Edition, John Wiley and sons, 2002.
4. M H Rashid, "Power Electronics, Circuits, Devices and Applications", Pearson, 3rd Edition, 2001.

V. WEB REFERENCES:

1. https://www.ee.iitkgp.ac.in
2. https://www.citchennai.edu.in
3. https://www.iare.ac.in

V Semester: EEE								
Course Code	Category	Hours / Week			Credits	Maximum Marks		
AECC31	Core	L	T	P	C	CIA	SEE	Total
		-	-	3	1.5	30	70	100
Contact Classes: Nil	Tutorial Classes: Nil	Practical Classes: 36				Total Classes: 36		
Prerequisites: Analog Electronics Digital Electronics								

I. COURSE OVERVIEW:

Microprocessors and Microcontrollers laboratory course helps the students to develop their knowledge on processor architecture and the programming skills. This laboratory course provides hands-on experience to interface I/O devices, perform A/D and D/A conversions, design traffic light etc.

II. COURSE OBJECTIVES:

The students will try to learn:

I. Develop assembly level programs and providing the basics of themicroprocessors.
II. Understanding the interfacing of external devices to the processor and controller for various applications.
III. Learn assemble language programming using 8051microcontrollers.
IV. Develop ability in programming using microprocessor and microcontroller.

III. COURSE SYLLABUS:

Expt. 1: DESIGN A PROGRAM USING WIN862

Design and develop an Assembly language program using 8086 microprocessor and to show the following aspects.

a) Programming
b) Execution
c) Debugging

To Demonstrate the win 862 software and Trainer kit for 8086 Microprocessor

Expt. 2: 16 BITARITHMETIC AND LOGICAL OPERATIONS

Write an ALP program to perform 16 Bit arithmetic and logical operations using WIN862 software

Expt. 3: MULTIBYTE ADDITION AND SUBRACTION

a) Write an ALP program to perform multi byte addition and subtraction
b) Write an ALP program to perform 3*3 matrix multiplication and addition

Expt. 4: PROGRAMS TO SORT NUMBERS

a) Write an ALP program to perform ascending order using 8086
b) Write an ALP program to perform descending order using 8086

Expt. 5: PROGRAMS FOR STRING MANIPULATIONS OPERATIONS

a) Write an ALP program to insert or delete a byte in the given string
b) Write an ALP program to search a number/character in a given string
c) Write an ALP program to move a block of data from one memory location to the othe
d) Write an ALP program for reverse of a given string.

Expt. 6: CODE CONVERSIONS

a) Write an ALP program to convert packed BCD to UnpackedBCD
b) Write an ALP program to convert packed BCD toASCII
c) Write an ALP program to convert hexadecimal toASCII

Expt. 7: INTERFACING STEPPER MOTOR

a) Write an ALP program to rotate stepper motor in clockwisedirection
b) Write an ALP program to rotate stepper motor in anti-clockwisedirection

Expt. 8: INTERFACING ADC & DAC DEVICES

a) Write an ALP program to convert analog to digital using8086

b) Write an ALP program to convert digital to analog using8086.

Expt. 9: INTERFACING KEYBOARD TO 8086

Write an ALP program to interface keyboard to 8086.

Expt. 10: SERIAL AND PARALLEL COMMUNICATION

a) Parallel communication between two microprocessors using 8255
b) Serial communication between two microprocessor kits using 8251.

Expt. 11: INTERFACING TRAFFIC LIGHT CONTROLLER AND TONE GENERATOR

a) Write a program to interface traffic light controller
b) Write an ALP program to interface tone generator.

Expt. 12: ARITHMETIC AND LOGICAL OPERATIONS USING 8051

Write an ALP program to perform 16 Bit arithmetic and logical operations using 8051 microcontrollers.

Expt. 13: TIMER/COUNTER

Write an ALP Program and verify Timer/Counter using 8051.

Expt. 14: INTERFACING KEYBOARD TO 8051

Write an ALP program to interface keyboard to 8051

IV. REFERENCE BOOKS:

1. Ray A.K, Bhurchandi K.M, "Advanced Microprocessor and Peripherals", TMH, 2nd Edition, 2012.
2. Muhammad AliMazidi, J.G.Mazidi and R.D McKinlay, "The 8051 Microcontroller and Embedded Systems using Assembly and C", Pearson Education, 2nd Edition, 2009.

V. WEB REFERENCES:

1. http://www.nptel.ac.in/downloads/106108100/
2. http://www.the8051microcontroller.com/web-references
3. http://www.iare.ac.in

OBJECT ORIENTED PROGRAMMING DEVELOPMENT AND LANGUAGES

V Semester: AE / ECE / EEE / ME / CE								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
ACSC23	**Skill**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		-	-	-	-	-	-	-
Contact Classes: Nil	**Total Tutorials: Nil**	**Total Practical Classes: Nil**				**Total Classes: Nil**		

I.COURSE OVERVIEW:

Java's unique architecture enables programmers to develop a single application that can run across multiple platforms seamlessly and reliably. This course, enable the students to gain extensive experience with Java and its object oriented features to create robust console and GUI applications and store and retrieve data from relational databases.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The basic object-oriented programming concepts and apply them in problem solving.
II. The inheritance concepts for reusing the program.
III. The programs to implement event handling, user interfaces and graphical interfaces with the help of Java.

III. COURSE SYLLABUS

MODULE-I: FUNDAMENTALS OF OBJECT-ORIENTED PROGRAMMING

Object oriented paradigm: Basic concepts of Object-Oriented Programming, Benefits of OOP, Applications of OOP; Java Evolution: Java Features, How Java differs from C and C++, Java and Internet,Java and World Wide Web, Web Browsers, Hardware and Software Requirements, Java Environment. Overview of Java Language: Simple Java Program, Java Program Structure, Java Tokens, Java Statements, Implementing a Java Program, Java Virtual Machine, Constants, Variables, Data types, Scope of Variables, Symbolic Constants, Type Casting and type promotions, Operators, Operator Precedence and Associativity, Control Statements, break, continue, Arrays-Multi dimensional arrays, Wrapper Classes, Simple examples.

MODULE-II: CLASSES AND OBJECTS

Classes and Objects, constructors, methods, this keyword, garbage collection, finalize, overloading methods and constructors, access control, static members, nested and inner classes, command line arguments, variable length arguments. Inheritance: Forms of inheritance, specialization, specification, construction, extension, limitation, combination, benefits and costs of inheritance. Super uses- final - polymorphism, method overriding - dynamic method dispatch, abstract classes, exploring String class.

MODULE-III: PACKAGES AND INTERFACES

Defining and accessing a package, understanding CLASSPATH, access protection importing packages, Interfaces: Defining and implementing an interface, Applying interfaces, Variables in interfaces and extended interfaces. Exploring java.lang and java.util packages.

Exception Handling: Fundamentals, usage of try, catch, multiple catch clauses, throw, throws and finally. Java Built in Exceptions and creating own exception subclasses.

MODULE- IV: MULTITHREADED PROGRAMMING

Java Thread life cycle model: Thread creation, Thread Exceptions, Thread Priority, Synchronization ,Messaging, Runnable Interface - Interthread Communication - Deadlock - Suspending, Resuming and stopping threads.
I/O Streams: File, Streams, Advantages, The stream classes, Byte streams, Character streams.

MODULE- V APPLET PROGRAMMING

Event handling: basics of event handling, Event classes, Event Listeners, delegation event model, handling mouse and keyboard events, adapter classes, AWT Class hierarchy, AWT Controls, Layout Managers and Menus, limitations of AWT. How Applets differ from Applications: Applet Life Cycle, Creating an Applet, Running the AppletDesigning a Webpage, Applet Tag, Adding Applet to HTML file, More about Applet Tag, Passing parameters to Applets, Aligning the display.

IV. TEXTBOOKS:

1. Herbert Schildt, "The Complete Reference Java J2SE", TMH Publishing Company Ltd, New Delhi, 5th Edition, 2008.
2. Cay Horstmann, "Big Java", John Wiley and Sons, 2nd Edition, 2006.

V. REFERENCES BOOKS:

1. H.M.Dietel and P.J.Dietel, "Java How to Program", Pearson Education/PHI, 6th Edition 2008.
2. Cay.S.Horstmann and Gary Cornell, "Core Java 2" Vol 1, Fundamentals", Pearson Education, 7th Edition, 2007.
3. Cay.S.Horstmann and Gary Cornell, "Core Java 2, Vol 2, Advanced Features", Pearson Education. 7th Edition, 2008.

VI. WEB REFERENCES:

1. http://www.javatpoint.com/java-tutorial
2. http://www.javatutorialpoint.com/introduction-to-java/

VII. E-Text Books:

1. http://bookboon.com/en/java-programming-language-ebooks
2. https://en.wikibooks.org/wiki/Java_Programming

POWER SYSTEM ANALYSIS

VI Semester: EEE								
Course Code	Category	Hours / Week			Credits	Maximum Marks		
AEEC22	Core	L	T	P	C	CIA	SEE	Total
		3	1	0	4	30	70	100
Contact Classes: 45	Tutorial Classes: 15	Practical Classes: Nil				Total Classes: 60		
Prerequisite: Electrical Power Generation System Electrical Power Transmission System								

I. COURSE OVERVIEW:

This course enables students to study the performance of interconnected power system under steady state and transient stability conditions. The course deals with formation of impedance and admittance matrices for various configurations, finding unknown electrical quantities at various buses, symmetrical and unsymmetrical fault analysis, power system using per unit representation.

II. COURSE OBJECTIVES:

Students will try to learn:

I. The per unit representation of electrical quantities and formation of network matrices for an interconnected power system network.
II. The load flow/power flow solution in a power system network using Gauss seidel, Newton Raphson, decoupled and fast decoupled techniques.
III. The use of thevinin's theorem and symmetrical components in short circuit fault analysis.
IV. The techniques to improve the power system stability under steady state and transient conditions.

III. COURSE SYLLABUS:

MODULE I- PER UNIT SYSTEM REPRESENTATION AND NETWORK MATRICES (10)

Per unit system representation: Single line diagram, reactance diagram of a three-phase power system.Network Matrices: Definitions of graph theory, bus incidence matrix, Y_{bus} formation by direct inspection and singular transformation methods; Formation of Z_{bus}: Partial network, algorithm for the modification of Z_{bus} matrix for addition of element from a new bus to reference bus, addition of element from a new bus to an old bus, addition of element between an old bus to reference bus and addition of element between two old busses, modification of Z_{bus} for the changes in network without mutual impedance, numerical Problems.

MODULE II-LOAD FLOW STUDIES (10)

Load flow studies: Necessity of power flow studies, classification of power system buses, derivation of static load flow equation, Load flow solution using Gauss Seidel method with and without PV buses, acceleration factor, algorithm and flowchart; Determination of bus voltages, injected active and reactive powers (Sample one iteration only) and finding line flows / losses for the given bus voltages; Load flow solution using Newton Raphson method in rectangular and polar coordinates form with or without PV busses, derivation of Jacobian elements, algorithm and flow chart; decoupled and fast decoupled methods, numerical problems.

MODULE III-SHORT CIRCUIT ANALYSIS (10)

Symmetrical fault analysis: short circuit (SC) in an unloaded synchronous machine, SC Fault current computation using reactance diagram and thevenin's theorem, SC MVA interrupting capacity of Circuit breaker, current limiting reactors, numerical problems.

Symmetrical components: Symmetrical component transformation, Sequence impedances and networks for transmission line, synchronous machine and transformer, sequence diagram of a power system, Unsymmetrical fault analysis: LG, LL, LLG faults with and without fault impedance, numerical problems.

MODULE IV-STEADY STATE STABILITY ANALYSIS (07)

Elementary concepts of steady state, dynamic and transient stabilities, dynamics of synchronous machine, swing equation, power flow under steady state, power angle equation and power angle curve, transfer reactance, steady state power limit, determination of steady state stability,synchronizing power coefficient and methods to improve steady state Stability, numerical problems.

MODULE V-TRANSIENT STABILITY ANALYSIS (08)

Transient stability by equal area criterion, application of equal area criterion to sudden changes in mechanical input, sudden loss of one of the parallel lines, sudden short circuit on one of the parallel lines, critical clearing angle and time, methods to improve transient stability, application of auto reclosing circuit breakers, numerical problems.

IV. TEXT BOOKS:

1. I J Nagrath& D P Kothari, “Modern Power system Analysis”, Tata McGraw-Hill Publishing Company, 2nd Edition.
2. M A Pai, “Computer Techniques in Power System Analysis”, TMH Publications.
3. B.R.Gupta, “Power System Analysis and Design”, S.CHAND publications
4. K Umarao, “Computer Techniques and Models in Power Systems”, I K International Pvt. Ltd.

V. REFERENCE BOOKS

1. Stagg , El Abiad, “Computer Methods In Power System”. Tata McGraw-Hill.1968.
2. Grainger and Stevenson, “Power System Analysis”, Tata McGraw-Hill, 3rd Edition, 2011.
3. J Duncan Glover and M S Sarma, “Power System Analysis and Design”, Thompson, 3rd Edition 2006.

VI. WEB REFERENCES

1. https://nptel.ac.in/courses/108/105/108105067/
2. https://www.coursera.org/learn/electric-power-systems
3. https://nptel.ac.in/content/storage2/courses/108104051/ui/Course_home-4.htm

VII.E-TEXT BOOKS

1. https://easyengineering.net/power-systems-analysis-by-grainger/
2. https://onlinelibrary.wiley.com/doi/book/10.1002/0471722901

ELECTRIC DRIVES AND STATIC CONTROL

<table>
<tr><td colspan="10">VI Semester: EEE</td></tr>
<tr><th>Course Code</th><th>Category</th><th colspan="3">Hours / Week</th><th>Credits</th><th colspan="3">Maximum Marks</th></tr>
<tr><td rowspan="2">AEEC23</td><td rowspan="2">Core</td><td>L</td><td>T</td><td>P</td><td>C</td><td>CIA</td><td>SEE</td><td>Total</td></tr>
<tr><td>3</td><td>1</td><td>0</td><td>4</td><td>30</td><td>70</td><td>100</td></tr>
<tr><td>Contact Classes: 45</td><td>Tutorial Classes: Nil</td><td colspan="4">Practical Classes: Nil</td><td colspan="3">Total Classes: 24</td></tr>
<tr><td colspan="10">Prerequisite: Power Electronics, DC Machines and Transformers, AC Machines</td></tr>
</table>

I. COURSE OVERVIEW

This course provides an introduction to the operation of electric drives controlled from a power electronic converter. This also deals with design concepts of controllers for motor drives. The development of compact thyristor power converters has made this possible by smooth speed control of both AC and DC motors which are employed for several applications such as AC drives, DC drives, vehicles and renewable energy.

II. OBJECTIVES:

The students will try to learn:

I. Operate and maintain different types of DC/AC and special electrical machine drives in the industry.
II. The steady state behavior and transient dynamics of the converter/chopper fed DC drive
III. The performance of different industrial drives considering issues such as energy efficiency, power quality, economic justification, environmental issues and practical liabilities.
IV. Starting, braking, and speed control arrangements for electric motors and their applications.

III. COURSE SYLLABUS:

MODULE-I: CONTROL OF DC MOTORS THROUGH PHASE CONTROLLED RECTIFIERS (12)

Introduction to thyristor controlled drives: Single phase semi and fully controlled converters connected to DC separately excited and dc series motors, continuous current operation, output voltage and current waveforms, speed and torque expressions, speed torque characteristics, problems on converter fed DC motors; Three phase semi and fully controlled converters connected to DC separately excited and DC series motors, output voltage and current waveforms, speed and torque expressions, speed torque characteristics and problems.

MODULE-II: SPEED CONTROL OF DC MOTORS (12)

Introduction to four quadrant operation: Motoring operations, electric braking, plugging, dynamic and regenerative braking operations; Four quadrant operation of DC motors by dual converters, closed loop operation of DC motor; Chopper fed DC drives: Single quadrant, two quadrant and four quadrant chopper fed DC separately excited and series excited motors, continuous current operation output voltage and current wave forms, speed torque expressions, speed torque characteristics, problems on chopper fed DC motors and closed loop operation.

MODULE-III: SPEED CONTROL OF INDUCTION MOTORS THROUGH VARIABLE VOLTAGE AND FREQUENCY (12)

Variable voltage characteristics: Control of induction motor by AC voltage controllers, waveforms, speed torque characteristics. Stepper motor and brushless DC motor drives.

Variable frequency characteristics: Variable frequency characteristics, variable frequency control of induction motor by voltage source and current source inverter and cycloconverters, pulse with modulation control, comparison of voltage source inverter and current source inverter operations, speed torque characteristics, numerical problems on induction motor drives, closed loop operation of induction motor drives.

MODULE-IV: SPEED CONTROL OF INDUCTION MOTORS THROUGH ROTOR RESISTANCE AND VECTOR CONTROL (12)

Static rotor Resistance control: Slip power recovery schemes, static Scherbius drive, static Kramer drive, their performance and speed torque characteristics, advantages and applications, vector control of induction motor drives: Principles of vector control, vector control methods, direct methods of vector control, indirect methods of vector control and problems.

MODULE-V: SPEED CONTROL OF SYNCHRONOUS MOTORS (12)

Separate control and self control of synchronous motors, operation of self controlled synchronous motors by voltage source inverter and current source inverter cyclo converters. Load commutated CSI fed synchronous motor, operation, waveforms, speed torque characteristics, applications, advantages and numerical problems, closed loop control operation of synchronous motor drives (block diagram only), variable frequency control, cycloconverter, PWM, variable frequency inverter and current source inverter.

IV. TEXT BOOKS:

1. PV Rao, “Power Semiconductor Drives”, BS Publications, 1st Edition, 2014.
2. G K Dubey, “Fundamentals of Electric Drives”, Narosa Publications, 2nd Edition, 2001.
3. SB Devan, GR Slemon, A Straughen, “Power semiconductor drives”, Wiley Pvt. Ltd,. 4th Edition, 2001.
4. B K Bose, “Modern Power Electronics and AC Drives”, Prentice Hall India Learning Private Limited, 2005

V. REFERENCEBOOKS:

1. Vedam Subramanyam, “Thyristor Control of Electric Drives”, Tata McGraw Hill Publication, 5th Edition, 2008.
2. John Hindmarsh, Alasdair Renfew”, Electrical machines and drive systems”, Oxford Butterworth Heinemann, 3rd Edition.
3. Austin Hughes, “Electrical motors and drives Fundamentals Types and Applications”, Elsevier, 3rd Edition, 2006.
4. M D Singh, K B Kanchandhani, “Power Electronics”, Tata McGraw Hill Publishing Company, 2nd Edition, 1998.
5. M H Rashid, “Power Electronics, Circuits, Devices and Applications”, Pearson, 3rd Edition, 2001
6. J. Gnanavadivel, “Power Semiconductor Drives”, Anuradha, 2nd Edition, 2007.

VI. WEB REFERENCES

1. https://www.electrical4u.com
2. https://www.iare.ac.in
3. https://www.researchgate.net
4. https://www.cusp.umn.edu

VII. E-TEXT BOOKS:

1. https://www.jntubook.com/
2. https://www.freeengineeringbooks.com

ELECTRICAL MEASUREMENTS AND INSTRUMENTATION

VI Semester: EEE								
CourseCode	Category	Hours/Week			Credits	Maximum Marks		
AEEC24	Core	L	T	P	C	CIA	SEE	Total
		3	0	0	3	30	70	100
Contact Classes:45	Tutorial Classes: Nil	Practical Classes: Nil			Total Classes:45			
Prerequisite: Electrical Circuits, DC Machines and Transformers								

I. COURSEOVERVIEW:

This course introduces and develops the basic understanding of measurement principles and measuring instruments used in numerous electrical applications. The course provides the concept of measurement, analysis of errors and various specification parameters used to judge and compare measuring instruments. It provides an insight to develop advanced instruments in industries.

II. COURSEOBJECTIVES:

The students will try to learn:

I. The types and characteristics of instruments employed for measuring electrical quantities.
II. The construction, operation and maintenance of different types of instruments.
III. The concepts of Cathode Ray Oscilloscope and transducers to measure the physical quantities in the field of science, engineering and technology.

III. COURSE SYLLABUS:

MODULE-I:INTRODUCTION TO MEASURING INSTRUMENTS (09)

Introduction: Classification of measuring instruments, deflecting, damping and control torques, types of errors, ammeter and voltmeter: PMMC, MI instruments, expression for deflection and control torque, errors and compensation, extension of range using shunts and series resistances; Electro static voltmeter. Attracted type, disc type, extension of range of voltmeters, electro dynamic type voltmeters.

MODULE-II:POTENTIOMETERS AND INSTRUMENT TRANSFORMERS (09)

Bloc DC Potentiometers: Principle and operation of Crompton potentiometer, standardization, measurement of unknown resistance, current, voltage; AC potentiometers: polar and coordinate type, standardization, applications; Instrument transformers: CT and PT, ratio and phase angleerror.

MODULE-III:MEASUREMENT OF POWER AND ENERGY (09)

Measurement of Power: Single phase dynamometer type wattmeter, LPF and UPF, double elements and three elements dynamometer wattmeter; Expression for deflection and control torque, extension of range of wattmeter by using instrument transformers, measurement of active and reactive power for balanced and unbalanced Systems.

Measurement of Energy: Single phase induction type energy meter, driving and braking torques, errors and compensations, testing by phantom loading using RSS meter, three phase energy meter, introduction to net energy metering (web ref: 4 and 5), maximum demand meters.

MODULE-IV:DC AND AC BRIDGES (9)

Measurement of Resistance: Methods of measuring low, medium, high resistance, Wheatstone bridge,Kelvin's double bridge, loss of charge method; Measurement of Inductance: Maxwell's bridge, Hay's bridge, Anderson's bridge, Owen's bridge; Measurement of Capacitance: Desauty's bridge, Wein's bridge, Schering bridge.

MODULE-V:TRANSDUCERS AND OSCILLOSCOPES (09)

Transducers: Definition of transducers, classification of transducers, advantages of electrical transducers,characteristics and choice of transducers, principle of operation of LVDT and capacitor transducers, LVDT applications, strain gauge and its principle of operation, gauge factor, thermistors, thermocouples, synchros, piezo-electric transducers, photovoltaic, photo conductive cells, photo diodes; Cathode ray oscilloscope: Cathode ray tube, electron projection in electric and magnetic fields, applications of CRO: measurement of phase and frequency, digital storage oscilloscope.

IV. TEXTBOOKS:

1. A K Sawhney, "Electrical and Electronic Measurement and Instruments", Dhanpat Rai and Sons Publications, 2002.
2. E W Golding and F C Widdis, "Electrical Measurements and Measuring Instruments", Wheeler Publishing, 5th Edition, 2006.

V. REFERENCE BOOKS:

1. Buckingham and Price, "Electrical Measurements", Prentice Hall.
2. D V S Murthy, "Transducers and Instrumentation", Prentice Hall of India, 2nd Edition, 2009.
3. A S Morris, "Principles of Measurement of Instrumentation", Pearson/Prentice Hall of India, 2nd Edition, 1994.
4. H S Kalsi, "Electronic Instrumentation", Tata McGraw-Hill Publications, 1st Edition 1995.

VI. WEBREFERENCES:

1. https://www.researchgate.net
2. https://www.aar.faculty.asu.edu/classes
3. https://www.facstaff.bucknell.edu/
4. https://www.electrical4u.com
5. https://www.iare.ac.in

VII. E-TEXTBOOKS:

1. https://www.jntubook.com
2. https://www.freeengineeringbooks.com
3. https://www.bookboon.com/en/mechanicshttps://www.freeengineeringbooks.com

DIGITAL CONTROL SYSTEMS

<table>
<tr><td colspan="8">VI Semester: AE</td></tr>
<tr><th>Course Code</th><th>Category</th><th colspan="3">Hours / Week</th><th>Credits</th><th colspan="3">Maximum Marks</th></tr>
<tr><td rowspan="2">AEEC25</td><td rowspan="2">Elective</td><td>L</td><td>T</td><td>P</td><td>C</td><td>CIA</td><td>SEE</td><td>Total</td></tr>
<tr><td>3</td><td>0</td><td>0</td><td>3</td><td>30</td><td>70</td><td>100</td></tr>
<tr><td>Contact Classes: 45</td><td>Tutorial Classes: Nil</td><td colspan="4">Practical Classes: Nil</td><td colspan="3">Total Classes: 45</td></tr>
<tr><td colspan="9">Prerequisite: Mathematical Transform Techniques , Control Systems</td></tr>
</table>

I. COURSE OVERVIEW:

This course deals with the basic concepts of Digital and Analog data control systems. It also deals with z- transforms and state space analysis. It elaborates the concept of stability and design of discrete time control system. This course addresses the various real time issues and the control strategies are used in communication areas.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The analog to digital and digital to analog conversion techniques.
II. The Z transforms techniques for solving difference equations.
III. How to apply state space analysis to determine the stability of digital control systems.
IV. Discrete time control system based on frequency responsemethod.

III. COURSESYLLABUS:

MODULE-I: SAMPLING AND RECONSTRUCTION (08)

Introduction examples of data control systems digital to analog conversion and analog to digital conversion, sample and hold operations.

MODULE-II: SYSTEM RESPONSE (10)

The z-transforms: Introduction, linear difference equations pulse response z-transforms theorems of z- transforms the inverse z-transforms modified z-transforms; Z-plane analysis of discrete time control system: Z-Transform method for solving difference equations pulse transforms function, block diagram analysis of sampled data systems mapping between s-plane and z-plane.

MODULE-III: STATE SPACE ANALYSIS (09)

State space representation of discrete time systems pulse transfer function matrix solving discrete time state space equations state transition matrix properties methods for computation of state transition matrix discretization of continuous time state space equations.

Controllability and observability: Concepts of controllability and observability tests for controllability and observability duality between controllability and observability controllability and observability conditions for pulse transfer function.

MODULE-IV: STABILITY ANALYSIS (10)

Mapping between the s-plane and z-plane primary strips and complementary strips constant frequency loci constant damping ratio loci stability analysis of closed loop systems in the z-plane Jury stability test stability analysis by the use of the bilinear transformation and Routh stability criterion.

MODULE-V: DESIG OF DISCRETE TIME CONTROL SYSTEM (08)

Design of discrete time control system by conventional methods: Transient and steady state response analysis design based on the frequency response method bilinear transformation and design procedure in the w plane lead lag and lead lag compensators and digital PID controllers; State feedback controllers and observers: Design of state feedback controller through pole placement necessary and sufficient conditions Ackerman' s formula, state observe rs, full order and reduced order observers.

IV. TEXTBOOKS:

1. K Ogata, “Discrete Time Control Systems”, Prentice Hall, 2nd Edition, 1995.
2. BC Kuo, “Digital Control Systems”, Oxford University Press, 2nd Edition, 2007.

3. Mgopal, "Digital Control and State Variable Methods", Tata McGraw-Hill, 2nd Edition, 2003.

V. REFERENCE BOOKS:

1. K Warwick, D Rees, "Industrial Digital Control Systems", Peter Peregrines Ltd, 2nd Edition, 1988.
2. K J Astroms and B. Wittenmark, "Computer Controlled Systems -Theory and Design", Prentice Hall, 3rd Edition, 2011.
3. Richard C Dorf Robert H. Bishop, "Modern Control Systems", Pearson Education inc, 1st Edition, 2008.

VI. WEBREFERENCES:

1. https://www.researchgate.net
2. https://www.aar.faculty.asu.edu/classes
3. https://www.facstaff.bucknell.edu/
4. https://www.electrical4u.com
5. https://www.iare.ac.in

VII. E-TEXTBOOKS:

1. https://www.jntubook.com/
2. https://www.freeengineeringbooks.com

PRINCIPLES OF SIGNALS AND SYSTEMS

VI Semester: AE								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
AEEC26	**Elective**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		3	-	-	3	30	70	100
Contact Classes: 45	**Tutorial Classes: Nil**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite: **Mathematical Transform Techniques, Control Systems**								

I. COURSE OVERVIEW:

This course is introducing the fundamental principles of signals and system analysis. These concepts form the building blocks of modern digital signal processing, communication and control systems. Hence, a sound understanding of these principles is necessary for all students of Electrical and Electronics Engineering

II. COURSE OBJECTIVES:

The students will try to learn:

I. The fundamentals about signals and systems.
II. The signals & systems (continuous and discrete) using time domain & frequency domain methods.
III. The stability of systems through the concept of ROC.
IV. The various transform techniques in the analysis of signals andsystems.

III. COURSE SYLLABUS:

MODULE – I: SIGNAL ANALYSIS (09)

Signal Analysis: Analogy between Vectors and Signals, Orthogonal Signal Space, Signal approximation using Orthogonal functions, Mean Square Error, Closed or complete set of Orthogonal functions, Orthogonally in Complex functions, Exponential and Sinusoidal signals, Concepts of Impulse function, Unit Step function, Signum function.

MODULE – II: FOURIER TRANSFORM (09)

Deriving Fourier Transform from Fourier series, Fourier Transform of arbitrary signal, Fourier Transform of standard signals, Fourier Transform of Periodic Signals, Properties of Fourier Transform

MODULE – III: SIGNAL TRANSMISSION THROUGH LINEAR SYSTEMS (09)

Linear System, Impulse response, Response of a Linear System, Linear Time Invariant (LTI) System, Linear Time Variant (LTV) System, Transfer function of a LTI system, Filter characteristics of Linear Systems, Distortion less transmission through a system, Signal bandwidth, System bandwidth, Ideal LPF, HPF and BPF characteristics

Convolution and Correlation of Signals: Concept of convolution in Time domain and Frequency domain, Graphical representation of Convolution, Convolution property of Fourier Transforms, Cross Correlation and Auto Correlation of functions, Properties of Correlation function, Relation between Convolution and Correlation.

MODULE- IV: INTRODUCTION TO DIGITAL SIGNAL PROCESSING (09)

Discrete Time Signals & Sequences, conversion of continuous to discrete signal, Normalized Frequency, Linear Shift Invariant Systems, Stability, and Causality, linear differential equation to difference equation, Linear Constant Coefficient Difference Equations, Frequency Domain Representation of Discrete Time Signals and Systems

MODULE- V: FAST FOURIER TRANSFORMS (09)

Fast Fourier transforms (FFT) - Radix-2 decimation-in-time and decimation-in-frequency FPT Algorithms, Inverse FFT and FFT with general Radix-N

IV. TEXT BOOKS:

1. B.P. Lathi, “Signals, Systems & Communications”, BS Publications, 2009.
2. A.V. Oppenheim, A.S. Willsky and S.H. Nawab, “Signals and Systems”, PHI, 2nd Edition 2009.

3. John G. Proakis, Dimitris G. Manolakis, “Digital Signal Processing, Principles, Algorithms, and Applications” Pearson Education / PHI. 2007.

V. REFERENCE BOOKS:

1. Simon Haykin and Van Veen, “Signals & Systems” Wiley, 2nd Edition, 2009.
2. Iyer and K. Satya Prasad, “Signals and Signals”, Cengage Learning, 2nd Edition, 2009.
3. A. V. Oppenheim and R.W. Schaffer, “Discrete Time Signal Processing”, PHI, 2009.
4. Loney Ludeman. John Wiley, “Fundamentals of Digital Signal Processing” PHI, 2009

VI. WEB REFERENCES:

1. https://www.edx.org/course/discrete-time-signal-processing-mitx-6-341x-1
2. https://www.mooc-list.com/course/digital-signal-processing-coursera

VII. E-TEXT BOOKS:

1. http://onlinevideolecture.com/ebooks
2. http://www.freebookcentre.net/SpecialCat/Free-Signal-Processing-Boo

EMBEDDED SYSTEMS AND IOT

V Semester: EEE								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
AEEC27	Elective	L	T	P	C	CIA	SEE	Total
		3	-	-	3	30	70	100
Contact Classes: 45	Tutorial Classes: Nil	Practical Classes: Nil				Total Classes: 45		
Prerequisite: Microprocessors and Microcontrollers								

I. COURSE OVERVIEW:

This course allows students to learn the fundamentals of embedded system hardware and firmware design and IoT environment. It focus on embedded system design process, embedded C, interfacing modules, software development tools , ARM & SHARC processor architectures , memory organization and IoT architectures and its tools. It provides hands-on experience on implementation of application prototype design in embedded and IoT environment.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The fundamental concepts of embedded computing, embedded C, RTOS and embedded software tools for implementing embedded systems.

II. Embedded software development tools for debugging and testing of embedded applications, architectures of ARM and SHARC processors.

III. Interfacing with external environments using sensors, actuators and communication in distributed embedded systems.

IV. The architecture of Internet of Things along with the usage of various hardware and sensing technologies to build IoT applications.

III. COURSE SYLLABUS:

MODULE-I: EMBEDDED COMPUTING (08)

Definition of embedded system, embedded systems vs. general computing systems, history of embedded systems, complex systems and microprocessor, classification, major application areas, the embedded system design process, characteristics and quality attributes of embedded systems, formalisms for system design, design examples

MODULE-II: INTRODUCTION TO ADVANCED PROCESSORS (08)

Introduction to advanced architectures: ARM and SHARC, processor and memory organization and instruction level parallelism; Networked embedded systems: Bus protocols, I2C bus and CAN bus; Internet-Enabled systems, design example-Elevator controller.

MODULE-III: INTRODUCTION TO EMBEDDED C AND APPLICATIONS (12)

C looping structures, register allocation, function calls, pointer aliasing, structure arrangement, bit fields, unaligned data and endianness, inline functions and inline assembly, portability issues; Embedded systems programming in C, binding and running embedded C program in Keil IDE, dissecting the program, building the hardware; Basic techniques for reading and writing from I/O port pins, switch bounce;

Applications: Switch bounce, LED interfacing, interfacing with keyboards, displays, D/A and A/D conversions, multiple interrupts, serial data communication using embedded C interfacing.

MODULE-IV: INTRODUCTION TO INTERNET OF THINGS AND M2M (09)

Definition and characteristics of IoT, physical design of IoT, logical design of IoT, IoT enabling technologies, IoT levels and deployment, domain specific IoTs. Introduction, M2M, difference between IoT and M2M, software defined networking (SDN) and network Function virtualization

(NFV) for IoT, basics of IoT system management with NETCONF-YANG.

MODULEV: IOT ARCHITECTURE AND TOOLS (08)

IoT Architecture: State of the art introduction, state of the art; Architecture reference model: Introduction, reference model and architecture, IoT reference model. IoT Reference model-IoT ecosystem and Business models- Introduction to Protocols of IoT: D2D, D2S, S2S, Introduction to simulation tools.

IV: TEXT BOOKS:

1. Sutton, Shibu K.V, "Introduction to Embedded Systems", Tata McGraw Hill Education Private Limited, 2nd Edition, 2009.
2. Raj Kamal, "Embedded Systems: Architecture, Programming and Design", Tata McGraw-Hill Education, 2nd Edition, 2011.
3. Arshdeep Bahga, Vijay Madisetti, Internet of Things: A Hands-on-Approach‖, VPT, 1st Edition, 2014.

V: REFERENCE BOOKS:

1. Wayne Wolf, "Computers as Components, Principles of Embedded Computing Systems Design", Elsevier, 2nd Edition, 2009.
2. Andrew Sloss, Dominic Symes, Wright, "ARM System Developer's Guide Designing and Optimizing System Software", 1st Edition, 2004.
3. Lyla B Das, "Embedded Systems", Pearson Education, 1st Edition, 2012.
4. Adrian McEwen, Hakim Cassimally,"Designing the Internet of Things", John Wiley and Sons, 1st Edition, 2014.

VI: WEB REFERENCES:

1. https://www.smartzworld.com/notes/embedded-systems-es/
2. http://notes.specworld.in/embedded-systems-es/
3. https://www.coursera.org/learn/iot.

VIII: E-TEXT BOOK:

1. https://www.scribd.com/doc/233633895/Intro-to-Embedded-Systems-by-Shibu-Kv
2. http://www.ee.eng.cmu.ac.th/~demo/think/_DXJSq9r3TvL.pdf
3. https://www.scribd.com/doc/55232437/Embedded-Systems-Raj
4. https://mitpress.mit.edu/books/internet-things

LINEAR SYSTEM ANALYSIS

VI Semester: AE								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
AEEC28	**Elective**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		3	-	-	3	30	70	100
Contact Classes: 45	**Tutorial Classes: Nil**	**Practical Classes: Nil**			**Total Classes: 45**			
Prerequisite: **Analog Electronics, Control Systems**								

I. COURSE OVERVIEW:

Develops the basic theory of continuous and discrete systems, with emphasis on linear time-invariant systems. Discusses the representation of signals and systems in both the time and frequency domain. Topics include linearity, time-invariance, causality, stability, convolution, system interconnection, and sinusoidal response. The Fourier and Laplace transforms are developed for the discussion of frequency domain applications. Sampling and quantization of continuous waveforms (A/D and D/A conversion) are analyzed, leading to the discussion of discrete-time FIR and IIR systems, recursive analysis, and realization. The Z-transform and the discrete-time Fourier transform are developed, and applied to the analysis of discrete-time signals and systems.

II. COURSE OBJECTIVES:

The students will try to learn:

I. Analyze linear systems and signals
II. Develop critical understanding of mathematical methods to analyze linear systems and signals.
III. Use mathematical modelling tools to represent linear systems

III. COURSE SYLLABUS:

MODULE-I: STATE VARIABLE ANALYSIS (09)

Choice of state variables in Electrical Networks-Formulation of state equations for Electrical networks Equivalent source method. Network topological method - Solution of state equations-Analysis of simple networks with state variable approach.

MODULE-II: FOURIER SERIES AND FOURIER TRANSFORM REPRESENTATION (09)

Introduction, Trigonometric form of Fourier series, Exponential form of Fourier series, Wave symmetry, Fourier integrals and transforms, Fourier transform of a periodic function, Properties of Fourier Transform, Parseval's theorem, Fourier transform of some common signals, Fourier transform relationship with Laplace Transform. Applications of Fourier series and Fourier Transform Representation: Introduction, Effective value, and average values of non-sinusoidal periodic waves, currents, Power Factor, Effects of harmonics, Application in Circuit Analysis, Circuit Analysis using Fourier Series.

MODULE-III: LAPLACE TRANSFORM APPLICATIONS (09)

Application of Laplace transform Methods of Analysis – Response of RL, RC, RLC Networks to Step, Ramp, and impulse functions, Shifting Theorem – Convolution Integral – Applications Testing of Polynomials: Elements of realizability - Hurwitz polynomials-positive real functions-Properties-Testing-Sturm's Test, examples.

Network Synthesis: Network synthesis: Synthesis of one port LC networks-Foster and CauerMethods-Synthesis of RL and RC one port networks-Foster and Cauer methods

MODULE-IV: SAMPLING (09)

Sampling theorem – Graphical and Analytical proof for Band Limited Signal impulse sampling, natural and Flat top Sampling, Reconstruction of signal from its samples, effect of under sampling – Aliasing, introduction to Band Pass sampling, Cross correlation and auto correlation of functions, properties of correlation function, Energy density spectrum, Power density spectrum, Relation between auto correlation function and Energy / Power spectral density function.

MODULE-V: Z-TRANSFORMS (09)

Fundamental difference between continuous and discrete time signals, discrete time complex, exponential and sinusoidal signals, periodicity of discrete time complex exponential, concept of Z Transform of a discrete sequence. Distinction between Laplace, Fourier, and Z-Transforms. Region of convergence in Z-Transforms, constraints on ROC for various classes of signals, Inverse Z-Transform properties of Z-Transforms.

IV. TEXT BOOKS:

1. B. P. Lathi", "Signals, Systems and Communications", BS Publications 2003.
2. Umesh Sinha, "Network Analysis and Synthesis", Satya Prakashan Publications, 2013.

V. REFERENCE BOOKS:

1. A. N. Tripathi, "Linear System Analysis", New Age International, 2nd Edition 1987.
2. D. Roy Chowdhary, "Network and Systems", New Age International, 2005.
3. Gopal G Bhise, Prem R. Chadha", Engineering Network Analysis and Filter Design, Umesh Publications 2009
4. A. Cheng, linear system analysis, Oxford publishers, 1999.

VI. WEB REFERENCES:

1. https://www.aar.faculty.asu.edu/classes.
2. https://www.books.askvenkat.com/engineering-textbooks/
3. https://www.electrical4u.com.

VII. E-TEXT BOOKS:

1. https://www.freebookcentre.net

COMPUTER ARCHITECTURE

OE – I: VI Semester: ECE / EEE OE –II: VII Semester: AERO / MECH / CIVIL								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
ACSC24	**Elective**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		3	-	-	3	30	70	100
Contact Classes: 45	**Tutorial Classes: Nil**	**Practical Classes: Nil**				**Total Classes: 45**		

I. COURSE OVERVIEW:

This course introduces the principles of computer organization and the basic architecture concepts. The main objective of this course is to give students to a clear understanding of the modern computer architecture. It also helps the students to know about hardware and software implementation of (ALU) arithmetic and logic unit to solve addition, subtraction, multiplication and division. It also defines the constituent parts of the system, how they are interconnected, and how they interoperate in order to implement the architectural specification. In this course, students will learn the basics of hardware components from basic gates to memory and I/O devices, instruction set architectures and designs to improve the performance.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The organization and architecture of computer systems and electronic computers.
II. The assembly language program execution, instruction format and instruction cycle.
III. How to design a simple computer using hardwired and micro programmed control methods.
IV. The basic components of computer systems besides the computer arithmetic.
V. The input-output organization, memory organization and management, and pipelining.

III. SYLLABUS

MODULE – I: INTRODUCTION TO COMPUTER ORGANIZATION (09)

Basic computer organization, CPU organization, memory subsystem organization and interfacing, input or output subsystem organization and interfacing, simple computer levels of programming languages, assembly language instructions, a simple instruction set architecture.

MODULE –II: ORGANIZATION OF A COMPUTER (09)

Register transfer: Register transfer language, register transfer, bus and memory transfers, arithmetic micro operations, logic micro operations, shift micro operations; Control memory.

MODULE –III: CPU AND COMPUTER ARITHMETIC (09)

CPU design: Instruction cycle, data representation, memory reference instructions, input-output, and interrupt, addressing modes, data transfer and manipulation, program control.

Computer arithmetic: Addition and subtraction, floating point arithmetic operations, decimal arithmetic unit.

MODULE –IV: INPUT-OUTPUT ORGANIZATION (09)

Input or output organization: Input or output Interface, asynchronous data transfer, modes of transfer, priority interrupt, direct memory access.

MODULE –V: MEMORY ORGANIZATION (09)

Memory organization: Memory hierarchy, main memory, auxiliary memory, associative memory, cache memory, virtual memory; Pipeline: Parallel processing, Instruction pipeline.

IV. TEXT BOOKS:

1. M. Morris Mano, “Computer Systems Architecture”, Pearson, 3rd Edition, 2015.
2. Patterson, Hennessy, “Computer Organization and Design: The Hardware/Software Interface”, Morgan Kaufmann, 5th Edition, 2013.

V. REFERENCE BOOKS:

1. John. P. Hayes, “Computer System Architecture”, McGraw-Hill, 3rd Edition, 1998.
2. Carl Hamacher, Zvonko G Vranesic, Safwat G Zaky, “Computer Organization”, McGraw-Hill, 5th Edition, 2002.
3. William Stallings, “Computer Organization and Architecture”, Pearson Edition, 8th Edition, 2010.

VI. WEB REFERENCES:

1. https://www.tutorialspoint.com/computer_logical_organization/
2. https://www.courseera.org/learn/comparch
3. https://www.cssimplified.com/.../computer-organization-and-assembly-language-programming

VI. E-TEXT BOOKS:

1. https://www.groupes.polymtl.ca/inf2610/.../ComputerSystemBook.pdf
2. https://www.cse.hcmut.edu.vn/~vtphuong/KTMT/Slides/TextBookFull.pdf

ADVANCED DATA STRUCTURES

<table>
<tr><td colspan="9">OE – I: VI Semester: ECE / EEE
OE –II: VII Semester: AERO / MECH / CIVIL</td></tr>
<tr><td>Course Code</td><td>Category</td><td colspan="3">Hours / Week</td><td>Credits</td><td colspan="3">Maximum Marks</td></tr>
<tr><td rowspan="2">ACSC25</td><td rowspan="2">Elective</td><td>L</td><td>T</td><td>P</td><td>C</td><td>CIA</td><td>SEE</td><td>Total</td></tr>
<tr><td>3</td><td>-</td><td>-</td><td>3</td><td>30</td><td>70</td><td>100</td></tr>
<tr><td>Contact Classes: 45</td><td>Tutorial Classes: Nil</td><td colspan="4">Practical Classes: Nil</td><td colspan="3">Total Classes: 45</td></tr>
</table>

I. COURSE OVERVIEW:

The course is intended to provide the foundations of the practical implementation and usage of Advanced Data Structures. It also covers some classical results and recent advancements on data structures, and the algorithms acting upon them. Typical topics include in sorting and searching, reorganizing lists and search trees based on the online sequence of queries to speed up searches, improving efficiency based on the distribution of queries, performing fast text retrieval by constructing indexes, and improving space efficiency of data structures for large data sets. The main objective of this course is to ensure that the student evolves into a competent programmer capable of designing and analyzing the implementations of different data structures for different kinds of problems.

II.COURSE OBJECTIVES:

The students will try to learn:

I. The basic data structures and techniques of algorithm analysis.
II. The dictionaries, hashing mechanisms and skip lists for faster data retrieval.
III. The comprehension of heaps, priority queues and its operations.
IV. Briefly about balanced trees and their operations.
V. The tries and pattern matching algorithms.

III. SYLLABUS:

MODULE – I: OVERVIEW OF DATA STRUCTURES (09)

Algorithms; Performance analysis: Time complexity and Space complexity, Asymptotic notation. Review of basic data structures - The list ADT, Stack ADT, Queue ADT, Linked list – Single linked list, Double linked list, Circular linked list.

MODULE – II: DICTIONARIES, HASH TABLES (09)

Dictionaries: Linear list representation, Skip list representation, operations - insertion, deletion and searching, Hash table representation, hash functions, collision resolution - separate chaining, open addressing - linear probing, quadratic probing, double hashing, rehashing, extendible hashing, comparison of hashing and skip lists.

MODULE – III: PRIORITY QUEUES (09)

Priority Queues – Definition, ADT, Realizing a Priority Queue using Heaps, Insertion, Deletion,.

Application-Heap Sort, External Sorting- Model for external sorting, Multiway merge, Polyphase merge.

MODULE – IV: SEARCH TREES (09)

Binary Search Trees - Definition, ADT, Operations - Searching, Insertion, Deletion, AVL Trees - Definition, ADT, Balance factor, Operations – Insertion, Deletion, Searching, Introduction to Red – Black and Splay Trees, B-Trees, B-Tree operations - insertion, deletion, searching, Comparison of Search Trees.

MODULE – V: PATTERN MATCHING AND TRIES (09)

Pattern matching algorithms - the Boyer - Moore algorithm, the Knuth – Morris - Pratt algorithm. Tries – Definition, concepts of digital search tree, Binary trie, Patricia, Multi-way trie.

IV. TEXT BOOKS:

1. Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, "Fundamentals of Computer Algorithms", Universities Press Private Limited, India, 2nd Edition, 2008.
2. G.A. V.Pai, "Data Structures and Algorithms", Tata McGraw Hill, New Delhi, 1st Edition, 2008.
3. Richard F Gilberg, Behrouz A Forouzan, "Data Structures - A Pseudocode Approach with C", Cengage Learning, Thomson Press (India) Ltd, 2nd Edition, 2006.

V. REFERENCE BOOKS:

1. D. Samanta, "Classic Data Structures", Prentice Hall of India Private Limited, 2nd Edition, 2003.
2. Aho, Hop craft, Ullman, "Design and Analysis of Computer Algorithms", Pearson Education India, 1st Edition, 1998.
3. Goodman, Hedetniemi, "Introduction to Design and Analysis of Algorithms", Tata McGraw Hill, New Delhi, India, 1st Edition, 2002.
4. Adam Drozdek, "Data Structures and Algorithms in C++", Thomson Course Technology, 3rd Edition, 2005.
5. M. T. Goodrich, R. Tomassia, "Data structures and Algorithms in Java", Wiley India, 3rd Edition, 2011.

VI. WEB REFERENCE:

1. https://www.tutorialspoint.com/data_structures_algorithms/data_structures_basics.htm
2. https://www.geeksforgeeks.org/data-structures/
3. http://www.nptelvideos.in/2012/11/data-structures-and-algorithms.html

VII. E-TEXT BOOKS:

1. https://pdfs.semanticscholar.org/19ec/55ed703eb24e1d98a4abd1a15387281cc0f8.pdf
2. https://www.academia.edu/35961658/Data.Structures.A.Pseudocode.Approach.with.C.2nd.edition_1_.pdf
3. https://sonucgn.files.wordpress.com/2018/01/data-structures-by-d-samantha.pdf

ARTIFICIAL INTELLIGENCE

OE – I: VI Semester: ECE / EEE OE –II: VII Semester: AERO / MECH / CIVIL							
Course Code	Category	Hours / Week			Credits	Maximum Marks		
ACSC26	Elective	L	T	P	C	CIA	SEE	Total
		3	-	-	3	30	70	100
Contact Classes: 45	Tutorial Classes: Nil	Practical Classes: Nil				Total Classes: 45		

I. COURSE OVERVIEW:

Artificial Intelligence has emerged as an increasingly impactful discipline in science and technology. AI applications are embedded in the infrastructure of many products and industries search engines, medical diagnoses, speech recognition, robot control, web search advertising and even toys. This course provides a broad overview of modern artificial Intelligence, learn how machines can engage in problem solving, reasoning, learning, and interaction design, test and implement algorithms.

II.COURSE OBJECTIVES:

The students will try to learn:

I. Gain a historical perspective of AI and its foundations.
II. Become familiar with basic principles of AI toward problem solving, inference, knowledge representation, and learning.
III. Explore the current scope, potential, limitations, and implications of intelligent systems.

III. SYLLABUS:

MODULE – I: INTRODUCTION (09)

Introduction: AI problems, Intelligent agents: Agents and Environments, the concept of rationality, the nature of environments, Structure of agents, Problem solving agents, Problem formulation.

MODULE – II: KNOWLEDGE REPRESENTATION & REASONS (09)

Knowledge – Based Agents, the Wumpus world.
Propositional Logic: Reasoning patterns in propositional logic - Resolution, Forward & Backward Chaining. Inference in First order logic: Propositional vs. first order inference.

MODULE – III: SEARCHING: (09)

Searching for solutions, uniformed search strategies – Depth limited search, bi-direction search, Comparing uninformed search strategies.

Search with partial information (Heuristic search), TSP problem, best first search, A* search, Hill climbing, Simulated annealing search.

MODULE – IV: CONSTRAIN SATISFACTION PROBLEMS (09)

Backtracking search for CSPs local search for constraint satisfaction problems. Game Playing: Games, Min - Max algorithm, Optimal decisions in multiplayer games, Alpha-Beta pruning.

MODULE – V: PLANNING: (09)

Classical planning problem, Language of planning problem, planning with state – space search, forward state spare search, backward state space search, Heuristics for state space search, Partial order planning Graphs, Planning graphs.

IV. TEXT BOOKS:

1. Stuart Russel, Peter Norvig, “Artificial Intelligence – A Modern Approach”, Pearson Education. 3rd Edition, 2009.

V. REFERENCE BOOKS:

1. E.Rich and K.Knight, “Artificial Intelligence”, Tata McGraw Hill, 3rd Edition, 2008.
2. Patterson, “Artificial Intelligence and Expert Systems”, PHI, 2nd Edition, 2009.
3. Giarrantana/ Riley, “Expert Systems: Principles and Programming”, Thomson, 4th Edition, 2004.
4. Ivan Bratka, “PROLOG Programming for Artificial Intelligence, Pearson Education, 3rd Edition, 2000.

CYBER CRIME AND COMPUTER FORENSICS

OE – I: VI Semester: ECE / EEE OE –II: VII Semester: AERO / MECH / CIVIL								
Course Code	Category	Hours / Week			Credits	Maximum Marks		
AITC19	Elective	L	T	P	C	CIA	SEE	Total
		3	-	-	3	30	70	100
Contact Classes: 45	Tutorial Classes: Nil	Practical Classes: Nil				Total Classes: 45		

I. COURSE OVERVIEW:

This course is designed to introduce the participant to the cybercrime prevention, detection and incident management processes, policies, procedures and cybercrime governance activities. The course is focus on cybercrime management standards, guidelines and procedures as well as the implementation and governance of these activities. In addition, it also provides the students an understanding of the new and advanced digital investigation techniques for machines, systems and networks since new technologies are opening today the door to new criminal approaches.

II.COURSE OBJECTIVES:

The students will try to learn:

I. The fundamental concepts of computer forensics and different types of forensics systems.
II. The methodologies to analyze and validate the forensics data.
III. The different tools and tactics that is associated with cyber forensics.

III. SYLLABUS:

MODULE – I: INTRODUCTION (09)

Introduction: Computer forensics fundamentals, types of computer forensics technology, types of computer forensics systems, vendor and computer forensics services.

MODULE – II: COMPUTER FORENSICS EVIDENCE AND CAPTURE (09)

Data recovery, evidence collection and data seizure, duplication and preservation of digital evidence, computer image verification and authentication.

MODULE – III: COMPUTER FORENSIC ANALYSIS (09)

Discover of electronic evidence, identification of data, reconstructing past events, fighting against macro threats.

Information warfare arsenal, tactics of the military, tactics of terrorist and rogues, tactics of private companies.

MODULE – IV: INFORMATION WARFARE (09)

Arsenal, surveillance tools, hackers and theft of components, contemporary computer crime, identity theft and identity fraud, organized crime &terrorism, avenues prosecution and government efforts, applying the first amendment to computer related crime, the fourth amendment and other legal issues.

MODULE – V: COMPUTER FORENSIC CASES (09)

Developing forensic capabilities, searching and seizing computer related evidence, processing evidence and report preparation, future issues.

IV. TEXT BOOKS:

1. John R. Vacca, "Computer Forensics: Computer Crime Scene Investigation", Cengage Learning, 2nd Edition, 2005. (UNIT I – IV)

2. Marjie T Britz, “Computer Forensics and Cyber Crime: An Introduction”, Pearson Education, 2nd Edition, 2008. (UNIT IV – V)

V. REFERENCE BOOKS:

1. MariE-Helen Maras, “Computer Forensics: Cybercriminals, Laws, and Evidence”, Jones & Bartlett Learning; 2nd Edition, 2014.
2. Chad Steel, “Windows Forensics”, Wiley, 1st Edition, 2006.
3. Majid Yar, “Cybercrime and Society”, SAGE Publications Ltd, Hardcover, 2nd Edition, 2013.
4. Robert M Slade, “Software Forensics: Collecting Evidence from the Scene of a Digital Crime”, Tata McGraw Hill, Paperback, 1st Edition, 2004.

ETHICAL HACKING

OE – I: VI Semester: ECE / EEE OE –II: VII Semester: AERO / MECH / CIVIL								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
AITC20	**Elective**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		3	-	-	3	30	70	100
Contact Classes: 45	**Tutorial Classes: Nil**	**Practical Classes: Nil**				**Total Classes: 45**		

I. COURSE OVERVIEW:

This course will provide fundamentals of the tools and techniques used by hackers and information security professionals alike to break into an organization. This course will immerse you into the Hacker Mindset so that you will be able to defend against future attacks. It puts you in the driver's seat of a hands-on environment with a systematic ethical hacking process. It will give an overview of how to scan, test, hack and secure own systems thought the different phases of ethical hacking include reconnaissance, gaining access, enumeration, maintaining access, and covering various tracks.

II.COURSE OBJECTIVES:

The students will try to learn:

I. The concepts of security testing and the knowledge required to protect against the hacker and attackers.
II. The reconnaissance and the publicly available tools used to gather information on potential targets.
III. The scanning techniques used to identify network systems open ports.
IV. The network system vulnerabilities and confirm their exploitability.
V. The techniques for identifying web application vulnerabilities and attacks.

III. SYLLABUS:

MODULE – I: INTRODUCTION TO HACKING (09)

Introduction to hacking, important terminologies, penetration test, vulnerability assessments versus penetration test, pre-engagement, rules of engagement, penetration testing methodologies, osstmm, nist, owasp, categories of penetration test, types of penetration tests, vulnerability assessment summary, reports.

MODULE – II: INFORMATION GATHERING AND SCANNING (09)

information gathering techniques, active information gathering, passive information gathering, sources of information gathering, tracing the location, traceroute, icmp traceroute, tcp traceroute, usage, udp traceroute, enumerating and fingerprinting the webservers, google hacking, dns enumeration, enumerating snmp, smtp enumeration, target enumeration and port scanning techniques, advanced firewall/ids evading techniques.

MODULE – III: NETWORK ATTACKS (09)

Vulnerability data resources, exploit databases, network sniffing, types of sniffing, promiscuous versus nonpromiscuous mode, mitm attacks, arp attacks, denial of service attacks.

Stripping https, traffic dns spoofing, arp spoofing attack manipulating the dns records, dhcp spoofing, remote exploitation, attacking network remote services, overview of brute force attacks, traditional brute force.

MODULE – IV: EXPLOITATION (09)

Introduction to metasploit, reconnaissance with metasploit, port scanning with metasploit, compromising a windows host with metasploit, client side exploitation methods, e–mails with malicious attachments, creating a custom executable, creating a backdoor with set, pdf hacking, social engineering toolkit, browser exploitation, post, exploitation, acquiring situation awareness, hashing algorithms, windows hashing methods.

MODULE – V: WIRELESS AND WEB HACKING (09)

Wireless hacking, introducing aircrack, cracking the wep, cracking a wpa/wpa2 wireless network using aircrack, ng – evil twin attack, causing denial of service on the original ap, web hacking, attacking the authentication, brute force and dictionary attacks, types of authentication.

IV. TEXT BOOKS:

1. Rafay Baloch, "Ethical Hacking and Penetration Testing Guide", CRC Press, 2014.

V. REFERENCE BOOKS:

1. Kevin Beaver, "Ethical Hacking for Dummies", Wiley, 6th Edition, 2018.
2. Jon Erickson , "Hacking: The Art of Exploitation", Rogunix, 2nd Edition, 2007.

MOBILE COMPUTING

OE – I: VI Semester: ECE / EEE **OE –II:** VII Semester: AERO / MECH / CIVIL								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
AITC21	**Elective**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		3	-	-	3	30	70	100
Contact Classes: 45	**Tutorial Classes: Nil**	**Practical Classes: Nil**				**Total Classes: 45**		

I. COURE OVERVIEW:

With the increasing popularity of mobile devices, mobile computing has become part of our daily life. This course will cover the nomenclature and implementation of mobile computing and mobile communication. It also provide a systematic explanation of mobile computing as a discrete discipline and will provide an in-depth coverage of mobile systems and devices used for application development, mobile databases, client-server computing agents, application servers, security protocols, mobile Internet, and ad-hoc and sensor networks.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The concept of wireless transmission protocols.
II. The typical mobile networking infrastructure through a popular GSM protocol architecture.
III. The various layers of mobile networks for location management.
IV. The database issues in mobile environments and data delivery models.
V. The platforms and protocols used in mobile environment.

III. SYLLABUS:

MODULE-I: INTRODUCTION (08)

Mobile computing – Paradigm, promises/Novel applications and impediments and architecture; Mobile and handheld devices, limitations of mobile and handheld devices. GSM – Services, system architecture, radio interfaces, protocols, localization, calling, handover, security, new data services, GPRS.

MODULE-II: MEDIA ACCESS LAYER AND MOBILE NETWORK LAYER (08)

Motivation for a specialized MAC (Hidden and exposed terminals. Near and far terminals), SDMA, FDMA, TDMA, CDMA, wireless LAN (IEEE802.11) system and protocol architecture; Mobile network layer: Packet delivery and handover management, location management, registration, tunneling and encapsulation, route optimization, DHCP.

MODULE-III: MOBILE TRANSPORT LAYER (08)

Conventional TCP/IP protocols, indirect TCP, snooping TCP, mobile TCP, other transport layers protocols for mobile networks;

Database issues: Database hoarding & caching techniques, C-S computing and adaptation, transactional models, query processing, data recovery process and QoS issues.

MODULE-IV: DATA DISSEMINATION AND SYNCHRONIZATION (10)

Communications asymmetry, classification of data delivery mechanisms, data dissemination, broadcast models, selective tuning and indexing methods.

MODULE-V: MOBILE ADHOC NETWORKS(MANET'S) (09)

Introduction, applications and challenges of a MANET, routing, classification of routing algorithms, algorithms such as DSR, AODV, DSDV; Mobile Agents, Service Discovery.

IV. TEXT BOOKS:

1. Jochen Schiller, “Mobile Communications”, Pearson Education, 2nd Edition, 2009.
2. Raj Kamal, “Mobile Computing”, Oxford University Press, Illustrated, 2nd Edition, 2012.

V. REFERENCE BOOKS:

1. Adelstein, Frank, Gupta, Sandeep KS, Richard III, Golden, Schwiebert, Loren, “Fundamentals of Mobile and Pervasive Computing”, McGraw-Hill Professional, 2005.
2. Hansmann, Merk, Nicklous, Stober, “Principles of Mobile Computing”, Springer, 2nd Edition, 2003.
3. Martyn Mallick, “Mobile and Wireless Design Essentials”, Wiley Dream Tech, 1st Edition, 2003.

VI. WEB REFERENCE:

1. https://en.wikipedia.org/wiki/Mobile_computing
2. https://www.tutorialspoint.com/mobile_computing/mobile_computing_quick_guide.h
3. https://media.techtarget.com/searchMobileComputing/downloads/Mobile_and_pervasive_computing_Ch 06pdf

VII. E-TEXT BOOKS:

1. https://books.google.co.in/books?id=HoFdSmH77wsC&printsec=frontcover&source=gbs_ge_summary_r&cad=0#v=onepage&q&false
2. https://books.google.co.in/books?id=LSqPLwEACAAJ&source=gbs_book_other_versions

EXPERIENTIAL ENGINEERING EDUCATION (ExEEd) – RESEARCH BASED LEARNING

VI Semester: Common for all branches

Course Code	Category	Hours / Week			Credits	Maximum Marks		
ACSC27	Foundation	L	T	P	C	CIA	SEE	Total
		2	-	-	1	30	70	100
Contact Classes: 36	Tutorial Classes: Nil	Practical Classes: Nil				Total Classes: 36		

Prerequisite: There are no prerequisites to take this course

I. COURSE OVERVIEW:

Research-based learning (RBL) presents as an alternative learning model that can develop the critical thinking skills. The research-based learning is conducted under constructivism which covers four aspects: learning which constructs student's understanding, learning through developing prior knowledge, learning which involves social interaction process, and meaningful learning which is achieved through real-world experience. The major focus is to engage students in the inquiry process where they formulate questions, conduct investigations, apply information and media to learning, and generate products that illustrate learning. The 5E learning cycle adopted for RBL leads students through five phases: Engage, Explore, Explain, Elaborate, and Evaluate which results in greater benefits concerning student's ability for scientific inquiry.

II. COURSE OBJECTIVES:

The students will try to learn:

I. To provide an opportunity for the students to engage in solving the real-world problems.
II. To introduce the overall process of research from its inception to the report.
III. To create the environment for multi-disciplinary research.
IV. Comprehend the role of ethics in research

III. COURSE SYLLABUS

I. What is Research?
II. Identifying Problem Statement
III. Overview of research-literature
IV. Planning activities, clarifying methods/methodologies
V. Experimentation
VI. Hypothesis testing
VII. Undertaking investigation and analyzing the data
VIII. Interpretation and consideration of results
IX. Presentation of replication studies

PLC AND INDUSTRIAL AUTOMATION LABORATORY

VI Semester: EEE								
Course Code	Category	Hours / Week			Credits	Maximum Marks		
AEEC32	Core	L	T	P	C	CIA	SEE	Total
		1	-	3	2.5	30	70	100
Contact Classes: 15	Tutorial Classes: Nil	Practical Classes: 36				Total Classes: 51		
Prerequisite: DC Machines and Transformers								

I. COURSE OVERVIEW:

The objective of this laboratory course is to measure the physical input variables and to analyze and control and output variables in an industrial automation process using programmable logic controllers (PLCs). The lab emphasizes on the software and hardware skills to design and realize an automation process. The lab is mainly intended to implement the software timers, counters and their usage in traffic signal control, sequential control, speed control of motors etc.

II. COURSE OBJECTIVES:

Students will try to learn:

I. The operation of PLCs, its ladder diagram programming and wiring of hardware equipment with PLC.
II. The measurement and control of digital, analog input/output variables using PLC.
III. The use of Human Machine Interface (HMI) to monitor and control the operation of a process.

III. COURSE SYLLABUS:

Module I: Introduction of PLC and Automation (15)

Introduction to Programmable Logic Controller (PLC), types of PLC, PLC selection, I/O devices and interfacing, digital and analog modules, communication modules, Basic Ladder logic, logic functions, electrical wiring diagram, scan cycle, timer-counter instructions, arithmetic functions, comparison functions, data handling, data move functions, introduction to Human Machine Interface(HMI) and interfacing HMI with PLC.

Module II: Laboratory Experiments (45)

Week – 1: LOGIC GATES

Implementation of logic gates using programmable logic controller.

Week – 2: TIMERS

Implementation of software timers in programmable logic controller.

Week – 3: COUNTERS

Implementation of software counters in programmable logic controller.

Week – 4: SEQUENTIAL CONTROL

Starting of three motors sequentially with some time delay using programmable logic controller.

Week – 5: DIRECT ONLINE (DOL) AND STAR DELTA STARTERS

Design of direct online (DOL) and star-delta starters for three phase induction motor using PLC.

Week – 6: AUTOMATIC FORWARD AND REVERSE CONTROL OF MOTOR

Automatic forward and reverse control of three phase squirrel cage induction motor for milling operation using programmable logic controller.

Week – 7: REVERSE CURRENT BREAKING OF THREE PHASE INDUCTION MOTOR

Implementation of reverse current braking method for three phase induction motor using PLC.

Week – 8: SPEED CONTROL OF DC MOTOR

Implementation of field control and armature control methods of speed control for DC motor using PLC.

Week – 9: WATER LEVEL MONITORING AND CONTROL

Development of automatic water level monitoring system for an overhead tank using PLC.

Week – 10: TRAFFIC SIGNAL CONTROL

Design of a traffic signal control system for a 3- way junction road using PLC.

Week – 11: TEMPERATURE CONTROL IN A ROOM

Design a temperature control system to monitor the temperature of a room using PLC.

Week – 12: OVER VOLTAGE AND UNDER VOLTAGE PROTECTION

Design of over voltage and under voltage protection system for home appliance using PLC.

IV. TEXT BOOKS:

1. Madhu Chanda Mitra, Samarjit Sen Gupta, "Programmable Logic Controllers and Industrial Automation: An Introduction", Penram International Publishing (India) Pvt. Ltd., 1st Edition, 2008.
2. K Krishnaswamy, S Vijayachitra, "Industrial Instrumentation", New Age Publications, 1st Edition, 2010.
3. Rajesh Mehra, Vikrant Vij, "PLCs & SCADA: Theory and Practice", Laxmi publications, 2nd Edition, 2016.
4. Stuart A. Boyer: SCADA-Supervisory Control and Data Acquisition, Instrument Society of America Publications, USA, 1999.

V. REFERENCE BOOKS

1. John R. Hack Worth, Frederick D. Hack Worth, Jr., "Programmable Logic Controllers: Programming Methods and Applications", Pearson Education, 4th Edition, 2008.
2. W. Bolton "Programmable logic controllers", Newnes Elsevier , 4th Edition, 2006.
3. Luis A. Bryan, E. A. Bryan, "Programmable Controllers theory and implementation", American Technical Publisher, 4th Edition, 2002.
4. Frank D. Petruzella, "Programmable Logic Controllers", Tata McGraw Hill, 3rd Edition, 2010.

VI. WEB REFERENCES

1. http://www.deltronics.ru/images/manual/DVP-ES2-EX2-SS2-SA2-SX2-SE-TP_PM_EN_20181030.pdf
2. https://downloadcenter.deltaww.com/downloadCenterCounter.aspx?DID=2077&DocPath=1&hl=en-US
3. https://www.freebookcentre.net/Electrical/PLC-Books.html
   https://library.automationdirect.com/plc-handbook/

ELECTRICAL MEASUREMENTS AND INSTRUMENTATION LABORATORY

IV Semester: EEE								
Course Code	Category	Hours / Week			Credits	Maximum Marks		
AEEC33	Core	L	T	P	C	CIA	SEE	Total
		0	0	3	1.5	30	70	100
Contact Classes: Nil	Tutorial Classes: Nil	Practical Classes: 36				Total Classes:36		
Prerequisite:There are no prerequisites to take this course.								

I. COURSE OVERVIEW:

The objective of this lab is to teach students to know the procedures for measuring Resistance, Inductance and Capacitance of different ranges. To perform experiments to measure three phase power and frequency. To design experiments for calibration of energy meter and power factor meter.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The various measurement techniques used in electrical engineering.
II. The different waveforms using LabVIEW to measure various parameters.
III. The use of sensors and transducers in electrical and nonelectrical measurements.
IV. The virtual instruments in measurement of analysis of electrical parameters.

III. COURSE SYLLABUS:

Week – 1: SENSING OF TEMPERATURE AND SPEED

Measurement of temperature using transducers like thermocouple, thermistors and resistance temperature detector with signal conditioning; speed measurement using proximity sensor using Hardware.

Week – 2:MEASUREMENT OF RESISTANCE

Measurement of low resistance using Kelvin's double bridge Hardware

Week – 3: MEASUREMENT OF STRAIN AND PRESSURE

Measurement of strain using straingauge and measurement of pressure using differential pressure transducer

Week – 4: MEASUREMENT OF POSITION AND LEVEL

Measurement of position using encoders and measurement of level using capacitive transducer

Week – 5: PHANTOM LOADING ON LPF WATTMETER

Calibration of e3lectrodynamometer type LPF wattmeter using phantom loading

Week – 6: CALIBRATION OF SINGLE PHASE ENERGY METER AND POWER FACTOR METER

Calibration of single phase energy meter using resistive load and dynamometer power factor meter

Week – 7: MEASUREMENT OF TURNS RATIO AND APPLICATIONS OF CTs

Measurement of turns ratio using AC bridge; the extension of range of wattmeter to measure three phase power using two CTs and one single phase wattmeter

Week – 8: MEASUREMENT OF REACTIVE POWER

Measurement of reactive power using one single phase wattmeter.

Week – 9: MEASUREMENT OF CAPACITANCE

Measurement of unknown capacitance using Schering bridge.

Week – 10:CROMPTON DC POTENTIOMETER

Calibration of PMMC ammeter and PMMC voltmeter.

Week – 11: ANALYSIS OF WAVE FORMS, FREQUENCY AND THD USING DIGITAL SIMULATION
Measurement and display of voltage, current wave forms, frequency Lissajeous patterns and THD using Lab VIEW.

Week – 12: MEASUREMENT OF THREE PHASE POWER
Measurement of three phase power with single wattmeter and two numbers of current transformer.

Week – 13: WORKING OF STATIC ENERGY METER USING DIGITAL SIMULATION
Measurement of energy using static energy meter and verification with Lab VIEW.

Week – 14: MEASUREMENT OF INDUCTANCE USING DIGITAL SIMULATION
Inductance measurement using Anderson bridge and verification with Lab VIEW.

IV. REFERENCE BOOKS:

1. https://www.bookpump.com/bwp/pdf-b/2335004b.pdf.
2. https://www.bambang.lecturer.pens.ac.id/rekayasa%20sensor%20aktuator/sensors%20&%20Trans.
3. https://www.sae.org/images/books/toc_pdfs/BELS036.pdf

V .WEB REFERENCES:

1. https://www.gnindia.dronacharya.info/EEEDept/Downloads/Labmanuals/EMI_Lab.pdf
2. https://www.scribd.com/doc/25086994/electrical-measurements-lab

DESIGN OF ALGORITHMS

<table>
<tr><td colspan="9">VI Semester: AE / ECE / EEE / ME / CE</td></tr>
<tr><th>Course Code</th><th>Category</th><th colspan="3">Hours / Week</th><th>Credits</th><th colspan="3">Maximum Marks</th></tr>
<tr><td rowspan="2">ACSC29</td><td rowspan="2">Skill</td><td>L</td><td>T</td><td>P</td><td>C</td><td>CIA</td><td>SEE</td><td>Total</td></tr>
<tr><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>Contact Classes: Nil</td><td>Total Tutorials: Nil</td><td colspan="4">Total Practical Classes: Nil</td><td colspan="3">Total Classes: Nil</td></tr>
</table>

I. COURSE OVERVIEW:

Design and analysis of algorithms is the process of finding the computational complexity of algorithms. It helps to design and analyze the logic on how the algorithm will work before developing the actual code for a program. It focuses on introduction to algorithm, asymptotic complexity, sorting and searching using divide and conquer, greedy method, dynamic programming, backtracking, branch and bound. NP-hard and NP-complete problems. The applications of algorithm design are used for information storage, retrieval, transportation through networks, and presentation to users.

II. COURSE OBJECTIVES:

The students will try to learn:

I. Assess how the choice of data structures and algorithm design methods impacts the performance of programs.
II. Solve problems using data structures such as binary search trees, and graphs and writing programs for these solutions.
III. Choose the appropriate data structure and algorithm design method for a specified application.
IV. Solve problems using algorithm design methods such as the greedy method, divide and conquer, dynamic programming, backtracking, and branch and bound and writing programs for these solutions.

III. SYLLABUS

MODULE –IINTRODUCTION

Algorithm: Pseudo code for expressing algorithms; Performance analysis: Space complexity, time complexity; Asymptotic notations: Big O notation, omega notation, theta notation and little o notation, amortized complexity; Divide and Conquer: General method, binary search, quick sort, merge sort, Strassen‘s matrix multiplication.

MODULE -IISEARCHING AND TRAVERSAL TECHNIQUES

Disjoint set operations, union and find algorithms; Efficient non recursive binary tree traversal algorithms, spanning trees; Graph traversals: Breadth first search, depth first search, connected components, biconnected components.

MODULE -IIIGREEDY METHOD AND DYNAMIC PROGRAMMING

Greedy method: The general method, job sequencing with deadlines, knapsack problem, minimum cost spanning trees, single source shortest paths.

Dynamic programming: The general method, matrix chain multiplication optimal binary search trees, 0/1 knapsack problem, single source shortest paths, all pairs shortest paths problem, the travelling salesperson problem.

MODULE -IVBACKTRACKING AND BRANCH AND BOUND

Backtracking: The general method, the 8 queens problem, sum of subsets problem, graph coloring, Hamiltonian cycles; Branch and bound: The general method, 0/1 knapsack problem, least cost branch and bound solution, first in first out branch and bound solution, travelling salesperson problem.

MODULE -VNP-HARD AND NP-COMPLETE PROBLEMS

Basic Concepts: Non-deterministic algorithms, the classes NP-Hard and NP-NP Hard problems, clique decision problem, chromatic number decision problem, Cook's theorem.

IV. TEXT BOOKS:

1. Ellis Horowitz, SatrajSahni, SanguthevarRajasekharan, “Fundamentals of Computer Algorithms”, Universities Press, 2nd Edition, 2015.
2. Alfred V. Aho, John E. Hopcroft, Jeffrey D, “The Design And Analysis Of Computer Algorithms”, Pearson India, 1st Edition, 2013.

V. REFERENCE BOOKS:

1. Levitin A, “Introduction to the Design and Analysis of Algorithms”, Pearson Education, 3rd Edition, 2012.
2. Goodrich, M. T. R Tamassia, “Algorithm Design Foundations Analysis and Internet Examples”, John Wileyn and Sons, 1st Edition, 2001.
3. Base Sara Allen Vangelder, “Computer Algorithms Introduction to Design and Analysis”, Pearson, 3rd Edition, 1999.

VI. WEB REFERENCES:

1. http://www.personal.kent.edu/~rmuhamma/Algorithms/algorithm.html
2. http://openclassroom.stanford.edu/MainFolder/CoursePage.php?course=IntroToAlgorithms
3. http://www.facweb.iitkgp.ernet.in/~sourav/daa.html

VII. E-TEXT BOOKS:

1. https://kailash392.files.wordpress.com/2019/02/fundamentalsof-computer-algorithms-by-ellis-horowitz.pdf.

POWER SYSTEM PROTECTION

VII Semester: EEE									
CourseCode	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
AEEC34	Core	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		3	1	0	4	30	70	100
Contact Classes:45	**Tutorial Classes:15**	**PracticalClasses:Nil**				**TotalClasses:60**		
Prerequisite: DC Machines and Transformers, Electrical Power Transmission Systems								

I. COURSEOVERVIEW:

The main objective of the course is to provide an overview of the principles and schemes for protecting power lines, transformers, buses, generators. It provides in depth knowledge of various types of relays and circuit breakers. It includes protection against over voltages in power system using lighting arrestors and insulation co-ordination

II. COURSEOBJECTIVES:

The students will try to learn:

I. The different types of circuit breakers.
II. Relays into various types such as of electromagnetic, static and numerical relays.
III. The performance of protection schemes of generator and transformer.
IV. The performance of feeder and bus-bar protection.
V. The protection schemes against over voltages.

III. COURSE SYLLABUS:

MODULE-I:CIRCUIT BREAKERS (08)

Circuit Breakers: Elementary principles of arc interruption, restriking and recovery voltages, restriking phenomenon, average, maximum and rate of rise of restriking voltage, current chopping and resistance switching, circuit breaker ratings and specifications, auto reclosures, description and operation of various types of circuit breakers, minimum oil circuit breakers, air blast circuit breakers, vacuum and SF6 circuit breakers, numerical problems.

MODULE-II:ELECTROMAGNETIC, STATIC AND NUMERICAL RELAYS (10)

Electromagnetic relays: Principle of operation and construction of attracted armature, balanced beam, induction disc and induction cup relays; Relays classification: instantaneous, definite minimum time and inverse definite minimum time relays over current / under voltage relays, direction relays, differential relays and percentage differential relays, universal torque equation; Distance relays: Impedance, reactance, mho and offset mho relays, characteristics of distance relays; Static relays: Overview of static relay, block diagram, operating principle and comparison, static relays versus electromagnetic relays; Numerical relays: Introduction, block diagram of numerical relay, sampling theorem, anti-aliasing filter, block diagram of phasor measurement unit and intelligent electronic device, data acquisition systems and numerical relaying algorithms, applications and numerical problems.

MODULE-III:SUBSTATIONS AND PROTECTION OF FEEDER / BUS BAR (09)

Indoor and outdoor substations: Substation's layout, bus bar arrangements like single, sectionalized, main and transfer bus bar system with relevant diagrams; Gas insulated substation (GIS): Types, single line diagram, constructional aspects of GIS, Installation, maintenance, advantages, comparison of GIS with air insulated substations.

Protection of lines: Over current, carrier current and three zone distance relay protection using impedance relays, translay relay; Protection of bus bars: Differential protection, grounded and ungrounded neutral systems, effect of ungrounded neutral on system performance, methods of neutral grounding, solid, resistance, reactance arcing grounds and grounding practices, application of numerical relays.

MODULE-IV:GENERATOR AND TRANSFORMER PROTECTION (10)

Generator protection: Protection of generators against stator faults, rotor faults, and abnormal conditions, restricted earth fault and inter turn fault protection, numerical problems on percentage winding unprotected; Transformer protection: Percentage differential protections, numerical problem on design of current transformers ratio, buchholz protection

MODULE-V:PROTECTION AGAINST OVER VOLTAGES (08)

Over voltages in power systems: Generation of over voltages in power systems, protection against lightning over voltages, valve type and zinc oxide lighting arresters, insulation coordination, basic insulation level, impulse ratio, standard impulse test wave, volt time characteristics.

IV. TEXTBOOKS:

1. Sunil S Rao, "Switchgear and Protection", Khanna Publishers, 1st Edition, 2013.
2. Badari Ram, D N Viswakarma, "Power System Protection and Switchgear", TMH Publications, 1st Edition, 2001.
3. A R van C Warrington, "Protective Relays: Their Theory and Practice", Springer Science & Business Media, Volume 2, 2nd Edition, 1977.
4. B L Soni, Gupta, Bhatnagar, Chakrabarthy, "Power System Engineering", Dhanpat Rai & Co, 3rd Edition, 2007.
5. T S Madhava Rao, "Power system protection: static relays", McGraw-Hill Companies, 2nd Edition, 1989.

V. REFERENCE BOOKS:

1. Paithankar, S R Bhide, "Fundamentals of Power System Protection", PHI, 1st Edition, 2003.
2. C L Wadhwa Electrical Power Systems", New Age international (P) Limited, 6th Edition, 2010.
3. VK Mehta," Principles of power systems", S Chand Publications, 4th Edition, 2009.

VI. WEB REFERENCES:

1. https://www.textbooksonline.tn.nic.in
2. https://www.freeengineeringbooks.com
3. https://www.eleccompengineering.files.wordpress.com
4. https://www.books.google.co.in

VII. E-TEXT BOOKS:

1. https://www.jntubook.com
2. https://www.freeengineeringbooks.com
3. https://www.bookboon.com/en/mechanicshttps://www.freeengineeringbooks.com

POWER SYSTEM OPERATION AND CONTROL

<table>
<tr><td colspan="10">VII Semester: EEE</td></tr>
<tr><th>Course Code</th><th>Category</th><th colspan="3">Hours / Week</th><th>Credits</th><th colspan="3">Maximum Marks</th></tr>
<tr><td rowspan="2">AEEC35</td><td rowspan="2">Core</td><td>L</td><td>T</td><td>P</td><td>C</td><td>CIA</td><td>SEE</td><td>Total</td></tr>
<tr><td>3</td><td>-</td><td>-</td><td>3</td><td>30</td><td>70</td><td>100</td></tr>
<tr><td>Contact Classes: 45</td><td>Tutorial Classes: Nil</td><td colspan="4">Practical Classes: Nil</td><td colspan="3">Total Classes: 45</td></tr>
<tr><td colspan="10">Prerequisite: DC Machines and Transformers, Power System Analysis</td></tr>
</table>

I. COURSE OVERVIEW:

This course enables students to analyze various parameters which effects the operation and control of power system. The course deals with control strategies to generate and distribute power in an interconnected system economically and reliably, maintenanceof the frequency and voltage within permissible limits, the factors influencing power generation at minimal cost are operating efficiencies, fuel cost and transmission losses, steady state flow of active and reactive power, development of optimal dispatch solutions to find the optimal dispatch of generation for an interconnected power system.

II. COURSE OBJECTIVES:

Students will try to learn:

I. The economic operation of power systems, hydro thermal scheduling.
II. The modeling of turbines, generators and automatic controllers.
III. The single area and two area load frequency control.
IV. Analyze reactive power control and load modeling.

III. COURSE SYLLABUS:

MODULE-I: ECONOMIC OPERATION OF POWER SYSTEMS (12)

Optimal scheduling of thermal power system: Optimal operation of generators in thermal power stations, heat rate curve, cost curve, incremental fuel and production costs, input output characteristics, optimum generation allocation without and with transmission line losses coefficients, general transmission line loss formula, unit commitment; Optimal scheduling of hydrothermal system: Hydroelectric power plant models, scheduling problems, short term hydro thermal scheduling problem.

MODULE-II: MODELING OF GOVERNOR, TURBINE AND EXCITATION SYSTEMS (09)

Modeling of governor: Mathematical modeling of speed governing system, derivation of small signal transfer function; Modeling of turbine: First order turbine model, block diagram representation of steam turbines and approximate linear models; Modeling of excitation system: Fundamental characteristics of an excitation system, transfer function, block diagram representation of IEEE type-1 model.

MODULE-III: SINGLE AREA AND TWO AREA LOAD FREQUENCY CONTROL (09)

Load frequency control of single area system: Necessity of keeping frequency constant, definitions of control area, single area control, block diagram representation of an isolated power system, steady state analysis, dynamic response, uncontrolled case.

Load frequency control of two area system: Uncontrolled case and controlled case, tie line bias control; Load frequency controllers: Proportional plus integral control of single area and its block diagram representation, steady state response, load frequency control and economic dispatch.

MODULE-IV: COMPENSATION FOR POWER FACTOR IMPROVEMENT AND REACTIVE POWER CONTROL (09)

Voltage control: Equipment for voltage control, effect of series capacitors, line drop compensation, effect of AVR, power factor control using different types of power capacitors, shunt and series capacitors, effect of shunt capacitors (fixed and switched), power factor correction, capacitor allocation, economic justification, procedure to determine the best capacitor location; Reactive power control: Reactive power compensation in transmission systems, advantages and disadvantages of different types of compensating equipment for transmission systems; Uncompensated and compensated transmission lines: Shunt and series compensation.

MODULE-V: LOAD COMPENSATION (06)

Load Compensation: characteristics of loads, factors associated with loads, relation between the load factor and loss factor; specifications of load compensator; Classification of loads: Residential, commercial, agricultural and industrial loads and characteristics.

IV. TEXT BOOKS:

1. C L Wadhwa, "Electrical power systems", New age International, 3rd Edition, 2005.
2. I J Nagarath, D P Kothari, "Modern power system analysis", Tata McGraw-Hill, 2nd Edition, 2006.

V. REFERENCE BOOKS:

1. Singh S N, "Electric Power Generation, Transmission and Distribution", Prentice Hall of India Pvt. Ltd., New Delhi, 2nd Edition, 2002.
2. T J E Miller, "Reactive power control in Electrical system", Wiley Interscience Publication, 1982.
3. V K Mehta and Rohit Mehta, "Principles of Power System", S Chand, 3rd revised Edition, 2015.
4. Turan Gonen, "Electrical Power Distribution System Engineering", CRC Press, 3rd Edition, 2014.
5. V Kamaraju, "Electrical Power Distribution Systems", TMH, Publication, Edition, 2009
6. O I Elgerd, "Electrical Energy Systems Theory", Tata McGraw-Hill, 2nd Edition, 2007.

VI. WEB REFERENCES:

1. https://www.electrical4u.com/working-or-operating-principle-of-dc-motor
2. https://www.freevideolectures.com
3. https://www.ustudy.in › ElectricalMachines
4. https://www.freeengineeringbooks.com

VII. E-TEXT BOOKS:

1. https://www.textbooksonline.tn.nic.in
2. https://www.freeengineeringbooks.com
3. https://www.eleccompengineering.files.wordpress.com
4. https://www.books.google.co.in

POWER SYSTEM STABILITY

VI Semester: EEE								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
AEEC36	**Elective**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		3	-	-	3	30	70	100
Contact Classes: 45	**Tutorial Classes: Nil**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite: DC Machines and Transformers, Power System Analysis								

I. COURSE OVERVIEW:

This course gives an insight into the various problems encountered in power systems related to reliability, stability and quality power. The course involves assessment of stability of a power system, improvement in stability and prevention of the system becoming unstable. The course would enable the students to figure out power system problems. This course deals with the development of detailed models of power system components and their application in the analysis of the dynamic behavior of interconnected power systems in response to small and large disturbances.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The state of estimation into different types.
II. How to monitor security and contingency evaluation.
III. The need of automation in power systems.
IV. The importance of voltage stability and voltage stability indices.
V. The artificial intelligence and artificial neural networks to power system analysis.

III. COURSE SYLLABUS:

MODULE-I: INTRODUCTION TO POWER SYSTEM STABILITY PROBLEMS (08)

Definition of stability, classification of stability, rotor angle stability, frequency stability, voltage stability, midterm and long-term stability, classical representation of synchronous machine in a single machine infinite bus system (SMIB), equal area criterion to asses stability of a single machine infinite bus system, limitations of classical model of synchronous machines.

MODULE -II: MODELING OF POWER SYSTEM COMPONENTS FOR STABILITY ANALYSIS (10)

Synchronous machine modeling: Sub transient model, two axis model, one axis (flux decay) model, classical model; Excitation systems modeling: DC excitation, AC excitation and static excitation, prime mover and energy supply systems modeling, transmission line modeling, load modeling, methods of representing synchronous machines in stability analysis.

MODULE -III: SMALL SIGNAL STABILITY (09)

Fundamental concepts, state space representation, modal analysis: Eigen properties, participation factors, stability assessment, effects of excitation system on stability.

Power system stabilizer and its design, angle and voltage stability of multi machine power systems and phenomenon of sub synchronous resonance.

MODULE -IV: TRANSIENT STABILITY (08)

Fundamentals of transient stability, numerical solution of algebraic differential equations, simultaneous implicit and partitioned explicit methods, analysis of unbalanced faults, direct method of transient stability, methods of improving transient stability.

MODULE -V: VOLTAGE STABILITY (08)

Classification of voltage stability, voltage stability analysis, modeling requirements, static and dynamic, prevention of voltage collapse.

IV. TEXT BOOKS:

1. P Kundur, "Power system Stability and Control", Tata McGraw Hill, 1st Edition, 2001.
2. M A Pai and Peter W Sauer, "Power System Stability", Pearson Education, 1st Edition, 2000.

V. REFERENCE BOOKS:

1. M A Pai, K Sengupta and K R Padiyar, "Topics on Small Signal Stability Analysis", Tata McGraw- Hill, 1st Edition, 2005.
2. K R Padiyar, "Power system dynamics", BSP Publications, 2nd Edition, 2010.
3. Paul M Anderson and A Fouad, "Power System Stability", Wiley-inter science, 1st Edition, 2002.

VI. WEB REFERENCES:

1.https://www.researchgate.net
2.https://www.aar.faculty.asu.edu/classes
3.https://www.facstaff.bucknell.edu/
4.https://www.electrical4u.com
5.https://www.iare.ac.in

VII. E-TEXT BOOKS:

1. https://www.jntubook.com/
2. https://www.freeengineeringbooks.com

POWER SYSTEM DYNAMICS AND CONTROL

VII Semester: EEE								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
AEEC37	**Elective**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		3	-	-	3	30	70	100
Contact Classes: 45	**Tutorial Classes: Nil**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite: DC Machines and Transformers, Control Systems, Power System Analysis								

I. COURSE OVERVIEW:

This course deals with the fundamental dynamic behavior and controls of power systems to perform basic stability analysis. Comprehend concepts in modeling and simulating the dynamic phenomena of power systems Interpret results of system stability studies. Analyze theory and practice of modeling main power system components, such as synchronous machines and excitation systems.

II. COURSE OBJECTIVES:

The students will try to learn:

I. Remember the dynamic characteristics of power system equipment,
II. Recognize dynamic performance of power systems
III. Illustrate the system stability and controls.

III. COURSE SYLLABUS:

MODULE-I: BASICCONCEPTS (9)

Power system stability states of operation and system security, system dynamics, problems system model analysis of steady, state stability and transient stability, simplified representation of excitation control.

MODULE-II: MODELINGOFSYNCHRONOUSMACHINE (10)

Synchronous machine, park‟s, Transformation – analysis of steady state performance, per–unit quantities, equivalent circuits of synchronous machine, determination of parameters of equivalent circuits.

MODULE-III: EXCITATIONSYSTEM (08)

Excitation system modeling, excitation systems, block diagram, system representation by state equations, dynamics of a synchronous generator connected to infinite bus, system model.

Synchronous machine model, stator equations rotor equations, synchronous machine model with field circuit, one equivalent damper winding on qaxis (model 1.1), calculation of Initial conditions.

MODULE-IV: ANALYSISOF SINGLEMACHINESYSTEM (09)

Small signal analysis with block diagram, representation characteristic equation and application of Routh Hurwitz criterion, synchronizing and damping torque analysis, small signal model, state equations.

MODULE-V: APPLICATION OF POWER SYSTEM STABILIZERS (09)

Basic concepts in applying PSS, control signals, structure and tuning of PSS, washout circuit, dynamic compensator analysis of single machine, infinite bus system with and without PSS.

IV. TEXT BOOKS:

1. K R Padiyar, “Power System Dynamics”, B S Publications, 1st Edition, 2001.
2. P M Anderson and A Fouad, “Power System Control and Stability”, IEEE Press, 2003.
3. P.Kundur, “Power System Stability and Control”, McGraw Hill Inc, New York, 1995.

V. REFERENCE BOOKS:

1. R Ramanujam, “Power Systems Dynamics”, PHI Publications, 1st Edition, 1998.
2. P.Sauer&M.A.Pai, “Power System Dynamics & Stability”, Prentice Hall, 1997.

VI. WEB REFERENCES:

1. https://www.eecs.umich.edu/eecs/pdfs/news/Hiskens598.pdf

2. https:// www.onlinecourses.nptel.ac.in/noc19_ee14/preview
3. http://www.infocobuild.com/education/audio-video

VII. E-Text Books:

1. https://courses.engr.illinois.edu/ece576/sp2018/Sauer%20and%20Pai%20book%20-%20Jan% 202007 .pdf.
2. https://www.researchgate.net/publication/41231911_Power_System_Dynamics_Stability_and_Contr ol

CONTROL SYSTEMS DESIGN

VII Semester: EEE									
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
AEEC38	**Elective**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		3	-	-	3	30	70	100
Contact Classes: 45	**Tutorial Classes: Nil**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite: Mathematical Transform Techniques, Control Systems								

I. COURSE OVERVIEW:

This course deals with the basic concepts of design specifications of time domain and frequency domain. It deals with various design of compensators and controllers in both time and frequency domain. It elaborates the concept of stability and its assessment for Nonlinear time invariant systems. This course address the various real time issues and how the control strategies are used in automation areas associates with variety of engineering streams

II. COURSE OBJECTIVES:

The students will try to learn:

I. The time and frequency domain design problem specifications.
II. The design aspects of classical control systems in frequency-domain.
III. The Design controllers to satisfy the desired design specifications using simple controller structures such as P, PI, PID, compensators.
IV. The performance of the systems by design them in state-space and study the effects of nonlinearities on various systems performance.

III. COURSE SYLLABUS:

MODULE-I: DESIGN SPECIFICATIONS (09)

Introduction to design problem and philosophy. Introduction to time domain and frequency domain design specification and its physical relevance. Effect of gain on transient and steady state response. Effect of addition of pole on system performance. Effect of addition of zero on system response.

MODULE-II: DESIGN OF CLASSICAL CONTROL IN THE TIME DOMAIN (09)

Introduction to compensator. Design of Lag, lead lag-lead compensator in time domain. Feedback and Feed forward compensator design. Feedback compensation. Realization of compensators.

MODULE-III: DESIGN OF CLASSICAL CONTROL SYSTEM IN FREQUENCY DOMAIN (09)

Compensator design in frequency domain to improve steady state and transient response. Feedback and Feed forward compensator design using bode diagram.

Design of PID Controllers: Design of P, PI, PD and PID controllers in time domain and frequency domain for first, second and third order systems. Control loop with auxiliary feedback — Feed forward control.

MODULE-IV: CONTROL SYSTEM DESIGN IN STATE SPACE (09)

Review of state space representation, concept of controllability and observability, effect of pole zero cancellation on the controllability and observability of the system, pole placement design through state feedback, Ackerman's formula for feedback gain design, design of observer, reduced order observer, separation principle.

MODULE-V: NONLINEARITIES AND ITS EFFECT ON SYSTEM PERFORMANCE (09)

Introduction to nonlinear systems, types of non-linearity's, Effect of various non-linearity's on system performance, introduction to phase plane analysis, singular points, phase plane analysis of nonlinear control systems.

IV. TEXT BOOKS:

1. I J Nagrath, M Gopal, "Control Systems Engineering", New Age International Publications, 1st Edition, 2007.
2. K Ogata, "Modern Control Engineering", Prentice Hall, 4th Edition, 2003.
3. N Nise, "Control System Engineering", John Wiley, 1st Edition, 2000.

V. REFERENCE BOOKS:

1. B C Kuo, "Automatic Control system", Prentice Hall, l" Edition, 1995.
2. J JD'Azzoand C. H. Houpis, "Linear Control System Analysis and Design (conventional and modem)", McGraw Hill, 1st Edition,1995.
3. RT Stefani and G.H.Hostetter, "Design of Feedback Control Systems", Saunders College Pub, 1st Edition, 1994.

VI. WEB REFERENCES:

1. https://www.electrical4u.com/working-or-operating-principle-of-dc-motor
2. https://www.freevideolectures.com
3. https://www.ustudy.in › ElectricalMachines
4. https://www.freeengineeringbooks.com

VII. WEB REFERENCES:

1. https://www.researchgate.net
2. https://www.aar.faculty.asu.edu/classes
3. https://www.facstaff.bucknell.edu/
4. https://www.electrical4u.com
5. https://www.iare.ac.in

VIII. E-TEXT BOOKS:

1. https://www.jntubook.com/
2. https://www.freeengineeringbooks.com

DIGITAL SIGNAL PROCESSING AND APPLICATIONS

VII Semester: EEE								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
AEEC39	**Elective**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		3	-	-	3	30	70	100
Contact Classes: 45	**Tutorial Classes: Nil**	**Practical Classes: Nil**				**Total Classes: 45**		
Prerequisite: Mathematical Transform Techniques, Control Systems								

I. COURSE OVERVIEW:

Thiscourseenablesthememorymanagementofsingleinstructionandmultipledata, very large instruction word and TMS processors for the implementation of discrete fourier transform and fast fourier transform algorithms. It focuses on memory organization, external bus interfacing signals, parallel input/output interface, interrupts, direct memory access, finite impulse response, infinite impulse response filters. In built peripherals of TMS processor used in applications of communication equipment, image processing, control systems and consumer electronic devices.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The architectures of digital signal processors and design aspects of digital signal processing algorithms.
II. The memory and external input/output peripheral interface with programmable DSP processor.
III. The realization of digital filters and fast fourier transform algorithms of the signal spectrum on host DSP processor.
IV. The programming skills using code composer studio environment for TMS320C54XX processor.

III. COURSE SYLLABUS:

MODULE - I:INTRODUCTION TO DIGITAL SIGNAL PROCESSOR (09)

Digital signal-processing and processors, Sampling process, Discrete time sequences, Discrete Fourier Transform (DFT) and fast Fourier transform (FFT), differences between DSP and other microprocessor architectures; Computational accuracy in DSP implementation-Number formats: Fixed point, floating point and block floating point formats, IEEE-754 floating point, dynamic range and precision, relation between data word size and instruction word size; Sources of error in DSP implementations: A/D conversion errors, DSP computational errors, D/A conversion errors.

MODULE - II: PROGRAMMABLE DIGITAL SIGNAL PROCESSORS (09)

Commercial Digital signal-processing Devices, Data Addressing modes of TMS320C54XX DSPs, Data Addressing modes of TMS320C54XX Processors, Memory space of TMS320C54XX Processors, Program Control, TMS320C54XX instructions and Programming, On-Chip Peripherals, Interrupts of TMS320C54XX processors, Pipeline Operation of TMS320C54XX Processors.

MODULE - III:STRUCUTRE OFIIR FILTERS (09)

Introduction, Analog filter approximations – Butterworth and Chebyshev, Design of IIR Digital filters from analog filters by Impulse invariant and bilinear transformation methods.

Frequency transformations, Basic structures of IIR Filters - Direct form-I, Direct form-II, Cascade form and Parallel form realizations.

MODULE - IV: SYMMETRIC AND ANTISYMMETRICFIR FILTERS (09)

Introduction, Characteristics of FIR filters with linear phase, Frequency response of linear phase FIR filters, Design of FIR filters using windows (Rectangular, Triangular, Raised Cosine, Hanning, Hamming, Blackman), Comparison of IIR & FIR filters, Basic structures of FIR Filters – Direct form, Cascade form, Linear phase realizations

MODULE - V: APPLICATIONS OF DSP (09)

Multirate signal processing; Decimation; Interpolation; Polyphase structures for decimation and interpolation filters; Structures for rational sampling rate conversion; Applications of multirate signal processing for design of phase shifters. Speech Signal Processing (Enhancement and Reproduction of speech signal), Image Processing (Steps in Digital Image Processing), Radar Signal Processing (Nonlinear and Nonstationary signal processing).

IV. TEXT BOOKS:

1. Avtar Singh and S. Srinivasan, “Digital Signal Processing”, Thomson Publications, 2004.
2. John G Proakis, G Manolakis, “Digital Signal Processing Principles, Algorithms, Applications,”, Prentice Hall India Private Limited, 4th Edition 2007.

V. REFERENCE BOOKS:

1. B. Venkata Ramani and M. Bhaskar, “Digital Signal Processors, Architecture, Programming and Applications”, TMH, 2004. (10th Print, 2007)
2. Byron Francis, “Raspberry PI 3: The Complete Beginner’s Guide”, Create Space Independent Publishing Platform, 2016.
3. Ifeachor E. C., Jervis B. W, “Digital Signal Processing: A Practical Approach”, Pearson Education, PHI/, 2nd Edition, 2002.
4. Peter Pirsch, “Architectures for Digital Signal Processing”, John Wiley, 1st Edition, 2007.

.

VI. WEB REFERENCES:

1. https://www.analog.com/en/design-center/landing-pages/001/beginners-guide-to-dsp.html
2. https://www.ti.com/microcontrollers-mcus-processors/processors/digital-signal-processors/overview.html
3. https://www.sciencedirect.com/topics/computer-science/digital-signal-processor
4. https://en.wikipedia.org/wiki/Digital_signal_processor

HVDC TRANSMISSION

VII Semester: EEE

Course Code	Category	Hours / Week			Credits	Maximum Marks		
AEEC40	Elective	L	T	P	C	CIA	SEE	Total
		3	-	-	3	30	70	100
Contact Classes: 45	Tutorial Classes: Nil	Practical Classes: Nil			Total Classes: 45			

Prerequisite: **Electrical Power Transmission Systems, Power System Analysis**

I. COURSE OVERVIEW:

This subject deals with the importance of HVDC transmission, analysis of HVDC Converters, Harmonics and Filters, Reactive power control and Power factor improvements of the system. It also deals with basic modeling and analysis of HVDC system power flow regulation.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The advantages of DC transmission over AC transmission.
II. The operation of Line Commutated Converters and Voltage Source Converters.
III. The analysis of control strategies used in HVDC transmission system.
IV.The stability improvement of power system using an HVDC system.

III. COURSE SYLLABUS:

MODULE-I: DC TRANSMISSION TECHNOLOGY (09)

Introduction to HVDC transmission systems: Comparison of AC and DC transmission (economics, technical performance and reliability), components of a HVDC system, types of HVDC links, application of DC transmission, Line Commutated Converter and Voltage Source Converter based systems.

MODULE-II:ANALYSIS OF LINE COMMUTATED AND VOLTAGE SOURCE CONVERTERS (09)

Line Commutated Converters (LCCs): Six pulse converter, analysis neglecting commutation overlap, harmonics, twelve pulse converters: Inverter operation, effect of commutation overlap, expressions for average DC voltage, AC current and reactive power absorbed by the converters, Effect of commutation failure, misfire and current extinction in LCC links; Voltage Source Converters (VSCs): two and three - level VSCs. PWM schemes: selective harmonic elimination, sinusoidal pulse width modulation, analysis of a six-pulse converter, equations in the rotating frame, real and reactive power control using a VSC.

MODULE-III: CONTROL OF HVDC CONVERTERS (09)

HVDC system control: Principles of link control in a LCC HVDC system, control hierarchy, firing angle controls: phase-locked loop, current and extinction angle control, starting and stopping of a link, higher level controllers power control, frequency control, stability controllers, reactive power control, principles of link control in a VSC HVDC system: power flow and DC voltage control, reactive power control, AC voltage regulation.

Components of HVDC systems: Smoothing reactors, reactive power sources and filters in LCC HVDC systems DC line, corona effects, insulators, transient over-voltages, DC line faults in LCC systems, DC line faults in VSC systems, DC breakers, monopolar operation, ground electrodes.

MODULE-IV: STABILITY ENHANCEMENT USING HVDC CONTROL (09)

Basic Concepts of stability enhancement: Power system angular, voltage and frequency stability, power modulation, basic principles, synchronous and asynchronous links, voltage stability problem in AC, DC systems.

MODULE-V: MTDC LINKS (09)

Introduction to MTDC links: Multi-terminal and multi in-feed Systems, series and parallel MTDC systems using LCCs, MTDC systems using VSCs, modern trends in HVDC Technology and introduction to modular multi-level converters.

IV. TEXT BOOKS:

1. K R Padiyar, "HVDC Power Transmission Systems: Technology and system Interactions", New Age International (P) Limited, 1st Edition, 1999.
2. S Rao, "EHVAC and HVDC Transmission Engineering and Practice", PHI, 3rd Edition, 1990.

V. REFERENCE BOOKS:

1. J Arrillaga, "High Voltage Direct Current Transmission", Peter Peregrinus Ltd., 1st Edition 1983.
2. E W Kimbark, "Direct Current Transmission", Vol.1, Wiley-Interscience, 1st Edition 1971.
3. E Uhlmann, "Power Transmission by Direct Current", B S Publications, 1st Edition, 1975.

VI. WEB REFERENCES:

1. https://www.as.wiley.com/WileyCDA/WileyTitle/productCd-1118634039.html.
2. https://www.springer.com/us/book/9788132221180.
3. https://www.springer.com/us/book/9781447151036.

VII. E TEXT BOOKS:

1. https://www.ijtra.com/view/role-of-power-electronics-in-non-renewable-and-renewableenergysystems.pdf.
2. https://www.nitgoa.ac.in/News_files/STC.pdf.
3. https://www.jee.ro/covers/art.php?issue=WN1438788776W55c22ca867606.

EHVAC TRANSMISSION

VII Semester: EEE								
CourseCode	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
AEEC41	**Core**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		3	0	0	3	30	70	100
Contact Classes:45	**Tutorial Classes:Nil**	**PracticalClasses:Nil**				**TotalClasses:45**		
Prerequisite: **Electrical Power Transmission System, Electric Drives and Static Control**								

I. COURSEOVERVIEW:

Modern power transmission is utilizing voltages between 345 kV and 1150 kV, A.C. Distances of transmission and bulk powers handled have increased to such an extent that extra high voltages and ultra-high voltages (EHV and UHV) are necessary. The problems encountered with such high voltage transmission lines exposed to nature are electrostatic fields near the lines, audible noise, radio interference, corona losses, carrier and TV interference and high voltage gradients. This course covers all topics that are considered essential for understanding the operation and design of EHV ac overhead lines and underground cables. Theoretical analyses of all problems combined with practical application are dealt in this course.

II. COURSEOBJECTIVES:

Thestudentswilltryto learn:

I. Illustrate basic concepts of extra high voltage AC transmission and understand the need for it.
II. Outline the line and ground reactive parameters and voltage gradients of conductors
III. Describe effects of corona and methods of associated measurement.
IV. Associate the knowledge of electro static field theory and traveling wave theory.
V. Select voltage control methods for extra high voltage AC transmission system.

III. COURSE SYLLABUS:

MODULE-I:INTRODUCTION TO EHVAC TRANSMISSION(08)

Necessity of EHV AC transmission – advantages and problems–power handling capacity and line losses- mechanical considerations – resistance of conductors – properties of bundled conductors – bundle spacing and bundle radius- Examples.

LINE AND GROUND REACTIVE PARAMETERS: Line inductance and capacitances – sequence inductances and capacitances – modes of propagation – ground return – Examples.

MODULE-II:VOLTAGE GRADIENTS CONDUCTORS(10)

Electrostatics – field of sphere gap – field of line changes and properties – charge – potential of relations for multi-conductors – surface voltage gradient on conductors – distribution of voltage gradient on sub-conductors of bundle – Examples.

MODULE-III:CORONA EFFECTS AND RADIO INTERFERENCE (09)

Corona Effects: Power loss and audible noise (AN) – corona loss formulae – charge voltage diagram – generation, characteristics – limits and measurements of AN – relation between 1-phase and 3-phase AN levels – Examples.

Radio Interference (RI) – corona pulses generation, properties, limits – frequency spectrum – modes of propagation – excitation function – measurement of RI, RIV and excitation functions – Examples.

MODULE-IV: ELECTRO STATIC FIELD(10)

Calculation of electrostatic field of EHV/AC lines – effect on humans, animals and plants – electrostatic induction in un-energized circuit of double-circuit line – electromagnetic interference-Examples.
Traveling wave theory: Traveling wave expression and solution- source of excitation- terminal conditions- open circuited and short-circuited end- reflection and refraction coefficients-Lumped parameters of distributed lines-generalized constants-No load voltage conditions and charging current.

MODULE-V:VOLTAGECONTROL(08)

Power circle diagram and its use – voltage control using synchronous condensers – cascade connection of shunt and series compensation – sub synchronous resonance in series capacitor – compensated lines – static VAR compensating system.

IV. TEXTBOOKS:

1. R. D. Begamudre, “Transmission Engineering” - New Age International (p) Ltd.
2. S. Rao, “EHV-AC, HVDC Transmission and distribution systems”- Khanna Publishers.
3. Padiyar K.R., “HVDC Power Transmission Systems” -New age International Ltd.

V. REFERENCEBOOKS:

1. Arrilaga, J, “High Voltage Direct Current Transmission”, peter pereginver Ltd., London, U.K.1983
2. Kimbark, E.W, “Direct Current Transmission-vol.1”, Wiley Interscience, New York, 1971

VI. WEBREFERENCES:

1. https://www.ae.pwr.wroc.pl/filez/20110606092353_HEV.pdf
2. https://www.afdc.energy pdfs/52723.pdf.gov/
3. https://www.leb.eei.uni- langen.de/winterakademie/2010/report/content/course03/pdf/0308.pdf
4. https://www.iare.ac.in

VII. E-TEXTBOOKS:

1. https://www.ae.pwr.wroc.pl/filez/20110606092353_HEV.pdf
2. https://www.afdc.energy.gov/pdfs/52723.pdf
3. langen.de/winterakademie/2010/report/content/course03/pdf/0308.pd

POWER ELECTRONICS IN RENEWABLE ENERGY SYSTEMS

VII Semester: EEE

Course Code	Category	Hours /Week			Credits	MaximumMarks		
AEEC42	Elective	L	T	P	C	CIA	SEE	Total
		3	-	-	3	30	70	100
Contact Classes:45	Tutorial Classes: Nil	Practical Classes: NIL				Total Classes: 45		

Prerequisite: Electrical Power Generating Systems, Power Electronics

I. COURSE OVERVIEW

This course highlights the applications of power electronics in renewable energy and power conversion systems. It describes the basic principles and characteristics of renewable energy sources, conversion systems and hybrid renewable energy resources in order to control and regulate the electrical power. The conversion and monitoring of electric energy with the use of semiconductors has been analyzed in this course elaborately.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The importance of energy conversion in the present energy scenario and the availability of renewable energy sources for sustainable energy conversion.

II. The basic concepts of power semiconductor devices to perform switching action in order to achieve a desired conversion strategy.

III. The hybrid renewable energy resources for grid integration and facilitate developing renewable systems for domestic and industrial applications.

III. COURSE SYLLABUS:

MODULE-I: INTRODUCTION TO RENEWABLE ENERGY SYSTEMS(09)

Environmental aspects of electric energy conversion: impacts of renewable energy generation on environment (cost-GHG Emission), qualitative study of different renewable energy resources ocean, biomass; Hydrogen energy systems: operating principles and characteristics of: Solar PV, fuel cells, wind electrical systems control strategy, operating area.

MODULE-II: ELECTRICAL MACHINES FOR RENEWABLE ENERGY CONVERSION (09)

Renewable energy conversion systems: fundamental principle of operation of self-excited induction generator, squirrel cage induction generator, doubly fed induction generator, synchronous generator, permanent magnet synchronous generator; Grid related problems: harmonic reduction and power factor improvement

MODULE-III: POWER CONVERTERS(09)

Solar: Block diagram of solar photo voltaic system, Line commutated converters (inversion mode), boost and buck-boost converters, selection of inverter, battery sizing, array sizing. Wind: three phase AC voltage controllers used in wind energy conversion.

Switching devices: AC – DC converters, PWM Inverters, Grid Interactive Inverters-matrix converters.

MODULE-IV: ANALYSIS OF WIND AND PV SYSTEMS(09)

Standalone operation of fixed and variable speed wind energy conversion systems and solar system, grid connection issues, grid integrated PMSG and SCIG Based WECS-Grid integrated solar system.

MODULE-V: HYBRID RENEWABLE ENERGY SYSTEMS(09)

Need for Hybrid Systems: Range and type of hybrid systems, case studies of wind-PV maximum power point tracking (MPPT), biomass-fuel cell hybrid systems, fuel cell-PV hybrid system and wind-fuel cell hybrid system

IV. TEXT BOOKS:

1. Mukund R Patel, “Wind and Solar Power Systems”, CRC Press, 1st Edition, 1999.
2. SN Bhadra, D. Kastha, S. Banerjee, “wind electrical systems”, OXFORD Higher Education, 2018.

V. REFERENCEBOOKS:

1. Daniel, Hunt. V, “Wind Power, A Hand Book of WECS”, Van Nostrend Co., Newyork, 2nd Edition, 1998.
2. G D Rai, “Non- Conventional Energy Resources”, Khanna Publishers, 1st Edition, 2002.
3. Arindam Ghosh, Gerard Ledwich, “Power Quality Enhancement Using Custom Power Devices”, Springer, 1stEdition, 2002.
4. Roger C Dugan, Mark E Mc. Granaghan, Surya Santosoh and H. Wayne Beaty, “Electrical Power Systems Quality”,

TATA McGraw Hill, 2nd Edition, 2010.

III. WEB REFERENCES

1. https://www.electrical4u.com
2. https://www.iare.ac.in
3. https://www.researchgate.net
4. https://www.cusp.umn.edu

IV. E-TEXT BOOKS:

1. https://www.jntubook.com/
2. https://www.freeengineeringbooks.com

WIND AND SOLAR ENERGY SYSTEMS

PE–IV

Course Code	Category	Hours /Week			Credits	MaximumMarks		
AEEC43	Elective	L	T	P	C	CIA	SEE	Total
		3	-	-	3	30	70	100
ContactClasses:45	TutorialClasses:Nil	PracticalClasses: NIL				TotalClasses: 45		

Prerequisite: Electrical Power Generating Systems

I. COURSE OVERVIEW

This course is intended to generation of electrical power using renewable energy sources like wind and solar. It describes power conditioning schemes for solar energy system, maximum power point tracking algorithms, wind energy conversion systems and power quality issues in the integration of renewable energy resources to grid.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The environmental aspects of renewable energy sources
II. The availability of renewable energy sources for sustainable conversion of energy
III. The impart knowledge on energy conversion systems in solar, wind and facilitate developing systems for different applications
IV. The present energy scenario and need for energy conversion
V. The power quality issues in integration of renewable energy resources

III. COURSE SYLLABUS:

MODULE-I: DESIGN AND OPERATION OF WIND POWER SYSTEM(09)

Wind Power System: Components, turbine rating, electrical load matching, variable-speed operation, system design features, maximum power operation, system control requirements, speed control, rate control and environmental aspects, wind energy conversion systems and their classification.

MODULE-II: DESIGN AND OPERATION OF PV SYSTEM (09)

Solar Photovoltaic Power System: The PV Cell, module and array, equivalent electrical circuit, open circuit voltage and short circuit current, I-V and P-V curves, array design, peak power point operation, PV system components; Solar Thermal System: Energy collection, synchronous generator, equivalent electrical circuit, excitation methods, electrical power output, transient stability limit, commercial power plants.

MODULE-III: POWER CONDITIONING SCHEMES FOR SOLAR ENERGY SYSTEMS(09)

Switching devices for solar energy conversion: DC power conditioning converters, maximum power point tracking algorithms.

AC Power conditioners, Line commutated inverters, synchronized operation with grid supply, Harmonic reduction.

MODULE-IV: WIND ENERGY CONVERSION SYSTEMS(09)

Wind energy Conversion system (WECS): Performance of Induction generators for WECS, Self-excited induction generator (SEIG) for isolated power generators. Controllable DC power from SEIGs, system performance, Grid related problems, generator control, AC voltage controllers, Harmonic reduction and Power factor improvement.

MODULE-V: POWER QUALITY ISSUES IN INTEGRATION OF RENEWABLE ENERGY RESOURCES (09)

Stand alone and Grid connected systems, Power Quality issues, Impact of power quality problems on DG, Mitigation of power quality problems, and Role of custom power devices in Distributed Generation.

IV. TEXT BOOKS:

1. Mukund R Patel, "Wind and Solar Power Systems", CRC Press, 1st Edition, 1999.
2. Publications, 2nd Edition, 2001. G D Rai, "Non- Conventional Energy Resources", Khanna Publishers, 1st Edition, 2002.

V. REFERENCEBOOKS:

1. Daniel, Hunt. V Wind Power, A Hand Book of WECS, Van Nostrend Co., Newyork, 2nd Edition, 1998.

2. ArindamGhosh, Gerard Ledwich, “Power Quality Enhancement Using Custom Power Devices”, Springer, 1st Edition, 2002.
3. Roger C Dugan, Mark E Mc. Granaghan, Surya Santosoh and H. Wayne Beaty, “Electrical Power Systems Quality”, TATA McGraw Hill, 2nd Edition, 2010.

III. WEB REFERENCES

1. https://www.electrical4u.com
2. https://www.iare.ac.in
3. https://www.researchgate.net
4. *https://www.cusp.umn.edu*

IV. E-TEXT BOOKS:

1. https://www.jntubook.com/
2. https://www.freeengineeringbooks.com

SOFT SKILLS AND INTERPERSONAL COMMUNICATION

OE – I: VI Semester: CSE / CSE (AI & ML) / CSE (DS) / CSE (CS) / CSIT / IT OE –II: VII Semester: ECE / EEE OE – III: VIII Semester: AERO / MECH / CIVIL								
Course Code	Category	Hours / Week			Credits	Maximum Marks		
AHSC15	Elective	L	T	P	C	CIA	SEE	Total
		3	-	-	3	30	70	100
Contact Classes: 45	Tutorial Classes: Nil	Practical Classes: Nil				Total Classes: 45		

I. COURSE OVERVIEW:

The objectives of the Soft Skills and Interpersonal Communication are to give each student a realistic perspective of work and work expectations, to help formulate problem solving skills, to guide students in making appropriate and responsible decisions, to create a desire to fulfill individual goals, and to educate students about unproductive thinking, self-defeating emotional impulses, and self- defeating behaviors.

II. COURSE OBJECTIVES:

The students will try to learn:

I. How to communicate in a comprehensible English accent and pronunciation.
II. The four language skills i.e., Listening, Speaking, Reading and Writing effectively.
III. The art of interpersonal communication skills to avail the global opportunities.
IV. The understanding of soft skills resulting in an overall grooming of the skills.

III. SYLLABUS

MODULE-I: SOFT SKILLS (09)

Soft Skills: An Introduction – Definition and Significance of Soft Skills; Process, Importance and Application of Soft Skills, Discovering the Self; Setting Goals; Positivity and Motivation: Developing Positive Thinking and Attitude.

MODULE –II: EFFECTIVENESS OF SOFT SKILLS (09)

Developing interpersonal relationships through effective soft skills; Define Listening, Speaking, Reading and Writng skills; Barriers to Listening, Speaking, Reading and Writing; Essential formal writing skills; Public Speaking: Skills, Methods, Strategies and Essential tips for effective public speaking.

MODULE-III: ORAL AND AURAL SKILLS (09)

Vocabulary:
Sounds of English vowels sounds and constant sounds, Word Accent and connected speech- contractions, questions tags, Listening for information, Taking notes while listening to lectures (use of Dictionary).

Group Discussion: Importance, Planning, Elements, Skills, Effectively disagreeing, Initiating.

MODULE-IV: VERBAL AND NON-VERBAL COMMUNICATION (09)

Interpersonal communication-verbal and nonverbal etiquette; Body language, grapevine, Postures, Gestures, Facial expressions, Proximity; Conversation skills, Critical thinking, Teamwork, Group Discussion, Impact of Stress; Measurement and Management of Stress.

MODULE-V: INTERPERSONAL COMMUNICATION (09)

Significance; Effectiveness of writing; Organizing principles of Paragraphs in documents; Writing introduction and conclusion; Techniques for writing precisely; Letter writing; Formal and Informal letter writing; E-mail writing, Report Writing.

IV. TEXT BOOKS:

Handbook of English for Communication (Prepared by Faculty of English, IARE)

V. REFERENCE BOOKS:

1. Dorch, Patricia. What Are Soft Skills? New York: Execu Dress Publisher, 2013.

2. Kamin, Maxine. Soft Skills Revolution: A Guide for Connecting with Compassion for Trainers, Teams, and Leaders. Washington, DC: Pfeiffer & Company, 2013.
3. Klaus, Peggy, Jane Rohman & Molly Hamaker. "The Hard Truth about Soft Skills", London: HarperCollins E-books, 2007.
4. Stein, Steven J. & Howard E. Book. "The EQ Edge: Emotional Intelligence and Your Success" Canada: Wiley & Sons, 2006
5. Suresh Kumar. English for Success. Cambridge University Press IndiaPvt.Ltd.2010.
6. Dorling Kindersley. Communication Skills & Soft Skills - An Integrated Approach. India Pvt. Ltd. 2013.

VI. WEB REFERENCES:

1. www.edufind.com
2. www.myenglishpages.com
3. http://grammar.ccc.comment.edu
4. http://owl.english.prudue.edu

VII. E-Text Books:

1. http://bookboon.com/en/communication-ebooks-zip
2. http://www.bloomsbury-international.com/images/ezone/ebook/writing-skills-pdf.pdf
3. https://americanenglish.state.gov/files/ae/resource_files/developing_writing.pdf
4. http://learningenglishvocabularygrammar.com/files/idiomsandphraseswithmeaningsandexamplespdf.pdf
5. http://www.robinwood.com/Democracy/General Essays/CriticalThinking.pdf

CYBER LAW AND ETHICS

<table>
<tr><td colspan="9">OE – I: VI Semester: CSE / CSE (AI & ML) / CSE (DS) / CSE (CS) / CSIT / IT
OE –II: VII Semester: ECE / EEE
OE –III: VIII Semester: AERO / MECH / CIVIL</td></tr>
<tr><td>Course Code</td><td>Category</td><td colspan="3">Hours / Week</td><td>Credits</td><td colspan="3">Maximum Marks</td></tr>
<tr><td rowspan="2">AHSC16</td><td rowspan="2">Elective</td><td>L</td><td>T</td><td>P</td><td>C</td><td>CIA</td><td>SEE</td><td>Total</td></tr>
<tr><td>3</td><td>-</td><td>-</td><td>3</td><td>30</td><td>70</td><td>100</td></tr>
<tr><td>Contact Classes: 45</td><td>Tutorial Classes: Nil</td><td colspan="4">Practical Classes: Nil</td><td colspan="3">Total Classes: 45</td></tr>
</table>

I. COUSRE OVERVIEW

This course consists of a sustained study of ethical and legal issues that arise in relation to employment in the public and private sectors, including allocation of resources, corporate and social responsibility, relationships, and discrimination. The main focus of this course will be on the ethical and legal standards governing information technology. New technology creates ethical challenges for individuals around the globe, and applies to most persons regardless of whether they are employed in the information technology field or a more traditional occupation. The study of this course provides a framework for making ethical decisions that professionals are likely to encounter in the workplace. This course will not only focus on ethics but on the legal, economic, social, cultural and global impacts of decisions that are made in the context of professional occupations.

II. COUSRE OBJECTIVES:

The students will try to learn:

I. The key terms and concepts in cyber society, cyber ethics.
II. The fundamentals of Cyber Law
III. The importance of nine P's in ethics.
IV. The artificial intelligence and Blockchain ethics.

III. SYLLABUS

MODULE-I: CYBER SOCIETY (09)

Definitions, Specificities of the Cyberspace, Dimensions of Cyber Ethics in Cyber Society, Fourth Industrial Revolution, Users' Motivations in Cyber-Space, Core Values and Virtues, Old Values or Eschatological Vision?, Cyber Ethics by Norms, Laws and Relations Artificial Intelligence Ethics: "AI for Good", Cyber-Capitalism: Cyber-Ethics as Business Ethics.

MODULE-II: CYBER LAW AND CYBER ETHICS (09)

Cyber Law and Cyber Ethics

The importance of cyber law, the significance of cyber ethics, cyber crime is unethical and illegal, ethics education has positive impact, the need for cyber regulation based on cyber ethics, very dangerous times.

MODULE-III: ETHICS IN THE INFORMATION SOCIETY, THE NINE P'S (09)

Principles: ethical values, participation: access to knowledge for all, people: community, identity, gender, generation, education, profession: ethics of information professions, privacy: dignity, data mining, security.

Piracy: intellectual property, cybercrime, protection: children and young people, power: Economic power of technology, media and consumers, policy: ethics of regulation and freedom.

MODULE-IV: DISRUPTIVE CYBER TECHNOLOGIES AND AI ETHICS (09)

Disruptive Cyber Technologies and Ethics -I

Artificial: negative moral judgment?, artificial: ethically positive innovation?, intelligence: action-oriented ability, creation story: human beings responsibility, the commandment to love and artificial intelligence;

Artificial Intelligence Ethics: Top nine ethical issues in artificial intelligence, five core principles to keep AI ethical, ethics should inform AI, but which ethics?

MODULE-V: DISRUPTIVE CYBER TECHNOLOGIES AND ETHICS –II (09)
Disruptive Cyber Technologies and Ethics -II
BLOCKCHAIN ETHICS:
Blockchain definition and description, Blockchain anonymity and privacy: ethical, no possibility to be forgotten, Blockchain for voting, Blockchain for transparent trade tracing, Blockchain energy: environmental impact, decentralized or majority-owned, ethically more benefits or dangers, future jobs in cyber society.

IV. TEXT BOOKS:

1. Christoph Stuckelberger, Pavan Duggal, "Cyber Ethics 4.0 Serving Humanity with Values", Globethics.net Global Series, 2018.

V. REFERENCE BOOKS:

1. Dr. Farooq Ahmad, Cyber Law in India, Allahbad Law Agency- Faridabad.
2. J.P. Sharma, SunainaKanojia, Cyber Laws
3. Harish Chander , Cyber Laws and IT Protection.

VI. E-REFERENCE:

https://www.globethics.net/documents/4289936/13403236/Ge_Global_17_web_isbn9782889312641.pdf/

ECONOMIC POLICIES IN INDIA

OE – I: VI Semester: CSE / CSE (AI & ML) / CSE (DS) / CSE (CS) / CSIT / IT **OE –II:** VII Semester: ECE / EEE **OE –III:** VIII Semester: AERO / MECH / CIVIL								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
AHSC17	**Elective**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		3	-	-	3	30	70	100
Contact Classes: 45	**Tutorial Classes: Nil**	**Practical Classes: Nil**				**Total Classes: 45**		

I. COURSE OVERVIEW

The objective of this course is to provide a broad sweep of the concept, structure and trends in the Indian economy in a roughly chronological manner. It begins with a review of the evolution of the Indian economy during colonial rule and introduces the roots of Indian underdevelopment. This course is designed to acquaint the students in a comprehensive manner with different aspects of Indian economy. The policy issues and measure to understand economic initiatives for improving economic development and growth, agriculture and industry, planning of the different sectors of the economy and the place of Indian economy in the international level particularly after economic reforms and covered.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The economic development elements and its measures
II. The inside knowledge on monetary policy and its importance in economic development
III. The importance of fiscal policies in promoting the economy
IV. The policies and practices in resource base infrastructure
V. The industrial and exit policies related to the industries

III. SYLLABUS

MODULE-I: INTRODUCTION ECONOMIC DEVELOPMENT AND ITS DETERMINANTS (09)

Approaches to economic development and its measurement – sustainable development; Role of State, market and other institutions; Indicators of development – PQLI, Human Development Index (HDI), gender development indices.

MODULE-II: MONEY, BANKING AND PRICES (09)

Analysis of price behavior in India; Financial sector reforms; Interest rate policy; Review of monetary policy of RBI; Money and capital markets; Working of SEBI in India.

MODULE-III: FISCAL POLICY AND PUBLIC FINANCES (09)

Fiscal federalism – Centre-State financial relations; Finances of central government; Finances of state governments; Parallel economy; Problems relating to fiscal policy; Fiscal sector reforms in India.

MODULE-IV: RESOURCE BASE AND INFRASTRUCTURE (09)

Energy; social infrastructure – education and health; Environment; Regional imbalance; Issues and policies in financing infrastructure development. Policies and Performance in Industry Growth; productivity; diversification; small scale industries; public sector; competition policy; foreign investment.

MODULE-V: THE INDUSTRIAL AND EXIT POLICIES (09)

Industrial policy; Public Sector enterprises and their performance; Problem of sick units in India; Privatization and disinvestment debate; Growth and pattern of industrialization; Small-scale sector; Productivity in industrial sector; Exit policy – issues in labour market reforms; approaches for employment generation.

IV. TEXT BOOKS:

1. The Wealth of Nations-Adam Smith, introduction by Alan B Krueger.
2. The Strength of Economic Development by Albert Hirschman.
3. Money, Banking and Public Finance by Dr. V.C.Sinha
4. Government of India, Economic Survey (Annual), Ministry of Finance, New Delhi.
5. Jain, a. K. (1986), Economic Planning in India, Ashish Publishing House, New Delhi.

V. REFERENCE BOOKS:

1. Ahluwalia, I. J. and I. M. D Little (Eds.) (1999), India's Economic Reforms and Development (Essays in honour of Manmohan Singh), Oxford University Press, New Delhi.
2. Bardhan, P. K. (9th Edition) (1999), The Political Economy of Development in India, Oxford University Press, New Delhi.
3. Bawa, R. s. and P. S. Raikhy (Ed.) (1997), Structural Changes in Indian Economy, Guru Nanak Dev University Press, Amritsar.
4. Brahmananda, P. R. and V. R. Panchmukhi (Eds.) (2001), Development Experience in the Indian Economy: Inter-State Perspectives, Book well, Delhi.
5. Chakravarty, S. (1987), Development Planning: The Indian Experience, Oxford University Press, New Delhi.
6. Dantwala, M. L. (1996), Dilemmas of Growth: The Indian Experience, Sage Publications, New Delhi.
7. Datt, R. (Ed.) (2001), Second Generation Economic Reforms in India, Deep & Deep Publications, New Delhi.

VI. WEB REFERENCE:

1. Parikh, K. S. (1999), India Development Report – 1999-2000, Oxford University Press, New Delhi8.
2. Reserve Bank of India, Report on Currency and Finance, (Annual).
3. Sandesara, J. c. (1992), Industrial Policy and Planning, 1947-19919 : Tendencies, Interpretations and Issues, Sage Publications, New Delhi.

GLOBAL WARMING AND CLIMATE CHANGE

<table>
<tr><td colspan="9">OE – I: VI Semester: CSE / CSE (AI & ML) / CSE (DS) / CSE (CS) / CSIT / IT
OE –II: VII Semester: ECE / EEE
OE –III: VIII Semester: AERO / MECH / CIVIL</td></tr>
<tr><td rowspan="3">Course Code

AHSC18</td><td rowspan="3">Category

Elective</td><td colspan="3">Hours / Week</td><td>Credits</td><td colspan="3">Maximum Marks</td></tr>
<tr><td>L</td><td>T</td><td>P</td><td>C</td><td>CIA</td><td>SEE</td><td>Total</td></tr>
<tr><td>3</td><td>-</td><td>-</td><td>3</td><td>30</td><td>70</td><td>100</td></tr>
<tr><td>Contact Classes: 45</td><td>Tutorial Classes: Nil</td><td colspan="4">Practical Classes: Nil</td><td colspan="3">Total Classes: 45</td></tr>
</table>

I. COURSE OVERVIEW

This course aims to address the whole complexity of climate change as an issue, by bringing together the science, impacts, economics, abatement technologies, and policy solutions. The course will address several important questions like what is the scientific basis for our understanding of climate change, and in what ways is that scientific basis uncertain. What changes in climate might we expect over the coming centuries? What would be the impacts of these changes in climate for human well-being and the natural world? What are the sources of emissions of greenhouse gases? What technologies exist or might be developed to allow us to slow climate change, and what international policy solutions might be necessary or preferred?

II. COURSE OBJECTIVES:

The students will try to learn:

I. The importance of Ozone layer in the atmosphere.
II. The comprehend composition of atmosphere.
III. The impacts of climate change on ecosystem.
IV. The initiatives taken by different countries to reduce emission of greenhouse gases.

III. SYLLABUS:

MODULE – I: EARTH'S CLIMATE SYSTEM (09)

Role of ozone in environment, Ozone layer – Ozone depleting gases, Green House Effect – Radioactive effects of Greenhouse gases, The Hydrological cycle, Green House Gases and Global Warming, Carbon Cycle.

MODULE –II: ATMOSPHERE AND ITS COMPONENTS (09)

Importance of Atmosphere – Physical and chemical characteristics of Atmosphere, Vertical structure of the atmosphere, Composition of the atmosphere, Atmospheric stability, Temperature profile of the atmosphere, Lapse rates, Temperature inversion, Effects of inversion on pollution dispersion.

MODULE – III: IMPACTS OF CLIMATE CHANGE (09)

Causes of Climate change: Changes of Temperature in the environment, Melting of ice pole, sea level rise, Impacts of Climate Change on various sectors – Agriculture, Forestry and Ecosystem, Water Resources, Human Health, Industry, Settlement and Society.

Methods and Scenarios, Projected Impacts for different regions, Uncertainties in the projected impacts of Climate Change, Risk of Irreversible Changes.

MODULE – IV: OBSERVED CHANGES AND ITS CAUSES (09)

Climate change and Carbon credits, CDM – Initiatives in India-Kyoto Protocol, Paris Convention - Intergovernmental Panel on Climate change, Climate Sensitivity and Feedbacks. The Montreal Protocol – UNFCCC – IPCC – Global Climate Models (GCM) - Evidences of Changes in Climate and Environment- on a Global scale and in India.

MODULE – V: CLIMATE CHANGE AND MITIGATION MEASURES (09)

Clean Development Mechanism, Carbon Trading – Examples of future clean technology, Biodiesel – Natural Compost, Eco-friendly plastic, Alternate Energy –Hydrogen, Bio-fules, Solar Energy, Wind and Hydroelectric Power. Mitigation Efforts in India and Adaptation funding. Key Mitigation Technologies and Practices –

Energy Supply, Transport, Buildings, Industry, Agriculture, Forestry – Carbon sequestration, Carbon capture and storage (CCS), Waste (MSW & Bio-waste, Biomedical, Industrial waste) – International and Regional cooperation.

IV. TEXT BOOKS:

1. Dr. Sushil Kumar Dash, "Climate Change: An Indian Perspective (Environment and Development)", Cambridge University Press India Pvt Ltd, 2007.
2. Adaptation and mitigation of climate change – Scientific Technical Analysis, Cambridge University Press, Cambridge, 2006.

V. REFERENCE BOOKS:

1. Atmospheric Science, J.M. Wallace and P.V Hobbs, Elsevier/ Academic Press, 2006.
2. "Climate Change and Climate Variability on Hydrological Regimes", Jan C. Van Dam, Cambridge University Press, 2003.

VI. E-TEXT BOOKS

1. https://www.worldcat.org/title/encyclopedia-of-global-warming-climate-change/oclc/805580328
2. https://libguides.nus.edu.sg/c.php?g=433566&p=2955835

INTELLECTUAL PROPERTY RIGHTS

OE – I: VI Semester: CSE / CSE (AI & ML) / CSE (DS) / CSE (CS) / CSIT / IT OE –II: VII Semester: ECE / EEE OE –III: VIII Semester: AERO / MECH / CIVIL								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
AHSC19	**Elective**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		3	-	-	3	30	70	100
Contact Classes: 45	**Tutorial Classes: Nil**	**Practical Classes: Nil**				**Total Classes: 45**		

I. COUSRE OVERVIEW:

The course will cover the philosophy of intellectual property rights, various technical and legal dimensions of IPR, and implications of IPR for growth and development of science, along with the various socio-economic and ethico-legal consequences of IPR on economic development. Students can also get disseminate knowledge on Design, Geographical Indication (GI), Plant Variety and Layout Design Protection and their registration aspects and also aware about current trends in IPR and Govt. steps in fostering IPR.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The knowledge in world trade organization and agreements between nations.
II. The intellectual property with international trade agreements.
III. The different types of intellectual property rights.
IV. The different laws in protection of intellectual property rights and its implementation.

III. SYLLABUS:

MODULE- I: INTRODUCTION (10)

General agreement on tariffs and trade (GATT) eight rounds: Uruguay round, world trade organization: structure, technology transfer, dispute resolution mechanism, Doha declaration world trade organization agreements including trade related intellectual properties rights and trade related investment measures.

MODULE- II: WORLD INTELLECTUAL PROPERTY ORGANIZATION (08)

Paris convention, Bern convention, Budapest treaty, Madrid agreement, huge agreement.

MODULE- III: PATENTS (09)

Historical background of intellectual property rights, introduction, definition and classification of intellectual property, patents, patentable and non-patentable inventions. Legal requirements for patents, types of patent applications.

Patent document: specification and claims, important procedural aspects, management of intellectual property rights assets and intellectual property portfolio, commercial exploitation of intellectual property.

MODULE- IV: DESIGNS AND GEOGRAPHICAL INDICATIONS (10)

Designs: basic requirements, procedure, convention application term, date, geographical indication: definition, what can be registered, who can apply, rights, term, restrictions.

MODULE- V: TRADEMARK AND COPYRIGHTS (08)

Definition, classification of trademarks, classifications of goods and services, Vienna classification, trademarks procedure, trademarks enforcement: infringement and passing off, remedies, copyrights, term of copyrights, and procedure of copyright assignment of copyright, copyright infringement remedies.

IV. TEXT BOOKS:

1. P. K. Vasudeva,World Trade Organization: Implications on Indian Economy, Pearson Education,2015.
2. P.KrishnaRao, WTO, Text and cases, Excel Books, 2015.
3. Carlos M.Correa- Intellectual property rights, The WTO and Developing countries-Zed books.

V. REFERENCE BOOKS:

1. Caves, Frankel, Jones, World Trade and Payments-An Introduction, Pearson4. Education, 2015.

2. Carlos M.Correa- Intellectual property rights, The WTO and Developing countries-Zed books.
3. Peter-Tobias stoll, Jan busche, Katrianarend- WTO- Trade –related aspects of IPR- Library of Congress.

VI. WEB REFERENCES:

1. http://www.ebooks directory.com
2. http://Campus guides.lib.utah.edu

VII. E-Text Books:

1. http://www.bookboon.com
2. http://www.freemagagement.com
3. http://www.emeraldinsight.com

ENTREPRENEURSHIP

OE – I: VI Semester: CSE / CSE (AI & ML) / CSE (DS) / CSE (CS) / CSIT / IT OE –II: VII Semester: ECE / EEE OE –III: VIII Semester: AERO / MECH / CIVIL								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
AHSC20	**Elective**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		3	-	-	3	30	70	100
Contact Classes: 45	**Tutorial Classes: Nil**	**Practical Classes: Nil**				**Total Classes: 45**		

I. COURSE OVERVIEW:

This course aims to provide students with an understanding of the nature of enterprise and entrepreneurship and introduces the role of the entrepreneur, will inculcate the knowledge of government supporting programs. Apart from this, students learn about the women entrepreneurs and success stories of women entrepreneurs, gain the knowledge of project management and profitability appraisal, focus on importance of training the new entrepreneurs as well as existing entrepreneurs. The students can also acquire necessary knowledge and skills required for organizing and carrying out entrepreneurial activities, for analysing and understanding business situations in entrepreneurs act and to master the knowledge necessary to plan entrepreneurial activities. The objective of the course is, to develop the ability of analysing various aspects of entrepreneurship – especially of taking over the risk, and the specificities as well as the pattern of entrepreneurship development and, finally, to contribute to their entrepreneurial and managerial potentials.

II. COUSRE OBJECTIVES:

The students will try to learn:

I. The Entrepreneurial process and also inspire them to be Entrepreneurs.
II. The key steps in the elaboration of business idea.
III. The stages of the entrepreneurial process and the resources needed for the successful development of entrepreneurial ventures.

III. SYLLABUS:

MODULE-I: UNDERSTANDING ENTREPRENEURIAL MINDSET (09)

The revolution impact of entrepreneurship- The evolution of entrepreneurship - Functions of Entrepreneurs – types of entrepreneurs -Approaches to entrepreneurship- Process approach- Role of entrepreneurship in economic development- Twenty first century trends in entrepreneurship.

MODULE-II: INDIVIDUAL ENTREPRENEURIAL MIND-SET AND PERSONALITY (09)

The entrepreneurial journey Stress and the entrepreneur - the entrepreneurial ego - Entrepreneurial motivations- Motivational cycle – Entrepreneurial motivational behavior – Entrepreneurial competencies. Corporate Entrepreneurial Mindset, the nature of corporate entrepreneur- conceptualization of corporate entrepreneurship Strategy-sustaining corporate entrepreneurship.

MODULE-III: LAUNCHING ENTREPRENEURIAL VENTURES (09)

Opportunities identification- Finding gaps in the market place – techniques for generating ideas- entrepreneurial Imagination and Creativity- the nature of the creativity process - Innovation and entrepreneurship.

Methods to initiate Ventures- Creating new ventures-Acquiring an Established entrepreneurial venture- Franchising- advantage and disadvantages of Franchising.

MODULE-IV: LEGAL CHALLENGES OF ENTREPRENEURSHIP (09)

Intellectual property protection - Patents, Copyrights - Trademarks and Trade secrets - Avoiding trademark pitfalls. Feasibility Analysis - Industry and competitor analysis - Formulation of the entrepreneurial Plan- The challenges of new venture start-ups, developing an effective business model – Sources of finance - Critical factors for new venture development - The Evaluation process.

MODULE-V: STRATEGIC PERSPECTIVES IN ENTREPRENEURSHIP (09)

Strategic planning - Strategic actions strategic positioning- Business stabilization - Building the adaptive firms - Understanding the growth stage – Internal growth strategies and external growth strategies, Unique managerial concern of growing ventures. Initiatives by the Government of India to promote entrepreneurship, Social and women entrepreneurship.

IV. TEXT BOOKS:

1. D F Kuratko and T V Rao, “Entrepreneurship- A South-Asian Perspective”, Cengage Learning, 2012.
2. Bruce R. Barringer/ R.Duane Ireland, “Entrepreneurship Successfully Launching New Ventures”, Pearson, 4th Edition, 2015.
3. S.S.Khanka, Entrepreneurship Development, S. Chand Publications, 2015.

V. REFERENCE BOOKS:

1. Stuart Read, Effectual Entrepreneurship, Routledge, 2013.
2. Rajeev Roy, Entrepreneurship, Oxford publications, 2nd Edition, 2012.
3. Nandan .H, Fundamentals of Entrepreneurship, PHI, 2013.

ELECTRICAL POWER SYSTEMS LABORATORY

VII Semester: EEE								
CourseCode	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
AEEC44	**Core**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		0	0	3	1.5	30	70	100
Contact Classes:45	**Tutorial Classes:Nil**	**Practical Classes: Nil**				**Total Classes:45**		
Prerequisite: Electrical Power Generating Systems, Electrical Power Transmission Systems								

I. COURSE OVERVIEW:

The main objective of the course is to provide an overview of the principles of basic protection circuits such as miniature circuit breaker, High rupturing fuse and protection under thermal overload condition. It provides in depth analysis of Ferranti effect and surge impedance loading of a transmission line. It provides in depth knowledge on working principles of various types of relays. It also deals with earth fault protection and feeder protection schemes

II. COURSE OBJECTIVES:

Thestudentswilltryto learn:

I. The importance of protection and plotting the characteristics of MCB and Fuse
II. The parameters, surge impedance loading and reactive power compensation of transmission lines
III. The concept of Ferranti effect of a transmission lines
IV. How to Calculate positive, negative and zero sequence impedances of synchronous machine

III. COURSE SYLLABUS:

LIST OF EXPERIMENTS

Week-1: CHARACTERISTICS OF AN MCB

Plotting the Characteristics of Miniature Circuit Breaker (MCB).

Week-2: CHARACTERISTICS OF FUSE AND THERMAL OVERLOAD PROTECTION

Study of characteristics of High Rupturing Capacity (HRC) fuse and tripping of bimetallic thermal overload protection and its characteristics.

Week-3: ABCD PARAMETERS OF TRANSMISSION LINE

Measurement of ABCD parameters of a transmission line

Week-4: FERRANTI EFFECT IN A TRANSMISSION LINE

Study of Ferranti effect in the transmission line

Week-5: SURGE IMPEDANCE LOADING

Study of Surge Impedance Loading (SIL) of a transmission line.

Week-6: EFFECT OF SHUNT COMPENSATION

Determine shunt compensation to counteract the voltage rise on no load and zero regulation at different loads in a transmission line.

Week-7: VOLTAGE PROFILE IMPROVEMENT USING TAP CHANGING TRANSFORMER

Study of voltage improvement by reactive power control using tap changing transformer.

Week-8: EFFICIENCY AND REGULATION OF A TRANSMISSION LINE

Determine the performance of a transmission line by calculating its efficiency and regulation.

Week-9: PERFORMANCE OF IMPEDANCE RELAY

Study the working principle of impedance relay and its effect during faults in a transmission line.

Week-10: PERFORMANCE OF OVER CURRENT RELAY
Study the working principle of over current relay and its effect during faults in a transmission line.
Week-11: EARTH FAULT PROTECTION
Study of earth fault detection methods and various earth fault protection schemes.

Week-12: FEEDER PROTECTION
Study the various protection schemes in radial feeder under various fault conditions.

Week-13: MEASURMENT OF SEQUENCE IMPEDANCES OF SYNCHRONOUS MACHINE
Measurement of positive, negative and zero sequence impedances of synchronous machine by using direct method and fault analysis method.

Week-14: STRING EFFICIENCY OF INSULATORS
Determination of string efficiency in a string of insulators.

IV. TEXTBOOKS:

1. Paithankar, S R Bhide, “Fundamentals of Power System Protection”, PHI, 1st Edition, 2003.
2. C L Wadhwa, “Electrical Power Systems”, New Age international (P) Limited, 6th Edition, 2010.
3. VK Mehta, “Principles of power systems”, S Chand Publications, 4th Edition, 2009.

V. REFERENCEBOOKS:

1. Badri Ram and D N Vishwakarma, “Power system Protection and Switchgear”, Tata McGraw-Hill Publication company limited, 1st Edition, 1995.
2. TS Madhava Rao, “Power system Protection static relay”, Tata McGraw-Hill Publishing Company limited, 2nd Edition, 1989.

POWER SYSTEM SIMULATION LABORATORY

VII Semester: EEE								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
AEEC45	**Core**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		0	0	3	1.5	30	70	100
Contact Classes: Nil	**Tutorial Classes: Nil**	**Practical Classes: 36**				**Total Classes:36**		
Prerequisite: Power System Analysis								

VI. COURSE OVERVIEW:

Power System simulation Laboratory comprises of protection, simulation, high voltage and machine related experiments. Varieties of Power system Simulation packages like Load flow, PSCAD and MATLAB are available.

VII. COURSE OBJECTIVES:

The students will try to learn:

I. Simulate transmission lines using PSCAD software to analyze faults in transmission system
II. The load flow studies using static load flow methods using MATLAB.
III. The transient state stability in power systems.

VIII. COURSE SYLLABUS:

Week – 1: FORMATION OF BUS ADMITTANCE AND IMPEDANCE MATRICES

Formation of bus admittance matrices by adding one element at a time and also write a program for Zbus building algorithm using MATLAB.

Week – 2: LOAD FLOW SOLUTION USING GAUSS SEIDEL METHOD

Write a MATLAB program for load flow studies without and with generator buses using Gauss Seidel Method.

Week – 3: LOAD FLOW SOLUTION USING NEWTON RAPHSON AND FDLF METHOD

Write a MATLAB program for load flow studies using Newton Raphson and Fast decoupled load flow (FDLF) method

Week – 4: POWER SYSTEM FAULT ANALYSIS

Analysis of symmetrical and unsymmetrical faults using symmetrical components using MATLAB.

Week – 5: POINT BY POINT METHOD

Development of MATLAB program for transient stability analysis of single machine, infinite bus and multi machine system by point by point method.

Week – 6: TRANSIENT RESPONSE OF RLC CIRCUIT

Obtain transient response of RLC circuit using PSCAD.

Week – 7: THREE PHASE SHORT CIRCUIT ANALYSIS IN A SYNCHRONOUS MACHINE

Analyze symmetrical faults and short circuit studies in a given synchronous machine using PSCAD.

Week – 8: STUDY OF TRANSMISSION SYSTEM AND SHORT CIRCUIT ANALYSIS OF 9 BUS SYSTEM

Study of simple transmission system and also Perform short circuit analysis on IEEE 9 bus system using PSCAD.

Week – 9: TRANSFORMER INRUSH CURRENT

Determination of transformer inrush current under unbalanced three phase parameters using PSCAD.

Week – 10: SMALL SIGNAL STABILITY ANALYSIS

Development of PSCAD / MATLABmodel for stability analysis of single machine - infinite bus with STATCOM.

Week – 11: TRANSMISSION LINE PARAMETERS

Obtaining parameters of a typical transmission line and modelling it in PSCAD.

Week – 12: LOAD FREQUENCY CONTROL

Obtain the frequency response of single and two area power system using PSCAD/MATLAB

Week – 13: POWER QUALITY

Familiarization with PSCAD and understanding of reactive power and power factor correction in AC circuits, current harmonics drawn by power electronics interface.

- Using the Network harmonic Impedance component
- Identifying network resonances
- Harmonic voltage amplification due to non linear loads

Week – 14: DISTANCE PROTECTION

Development of PSCAD model to study the distance protection scheme in long transmission line.

IX. REFERENCE BOOKS:

1. M A Pai, "Computer Techniques in Power System Analysis", TMH Publications, 1st Edition, 2010
2. Grainger, Stevenson, "Power System Analysis", Tata McGraw-Hill, 1st Edition, 2010.
3. Badri Ram and D N Vishwakarma, "Power System Protection and Switchgear", Tata McGraw-Hill Publication company limited, 1st Edition, 1995.
4. Paithankar, S R Bhide, "Fundamentals of Power System Protection", PHI, 1st Edition, 2003.
5. C L Wadhwa, "Electrical Power Systems", New Age international (P) Limited, 6th Edition, 2010.

X. WEB REFERENCES:

1. https://www.ee.iitkgp.ac.in
2. https://www.iare.ac.in /

HIGH VOLTAGE ENGINEERING

VIII Semester: EEE

Course Code	Category	Hours / Week			Credits	Maximum Marks		
AEEC47	Elective	L	T	P	C	CIA	SEE	Total
		3		-	3	40	60	100
Contact Classes: 45	Tutorial Classes: Nil	Practical Classes: Nil				Total Classes: 60		

Prerequisites: Electro Magnetic Fields

I. COURSEOVERVIEW:

This course enables Planning, operation and Testing of High voltage Electrical devices. High voltage engineering deals with different mediums of insulation and break down Phenomenon, generation of high DC and AC voltage, measurement Techniques of high AC and DC voltages, testing of insulation under all types of conditions using generated high DC and AC voltages.

II. COURSE OBJECTIVES: The students will try to learn:

I. The breakdown phenomena in gas, liquid and solid dielectric materials used in the high voltage devices.
II. The circuit design and operation for generation of high DC, AC and impulse voltages.
III. The different methods for measurement and testing of equipment's used in the high voltage engineering.

III. COURSE SYLLABUS:

MODULE-I: OVER VOLTAGES IN ELECTRICAL POWER SYSTEMS (9)

Origin of over voltages: Causes of over voltages and their effects on power system, lightning, switching surges and temporary over voltages, corona and its effects, reflection and refraction of travelling waves, protection against over voltages, charge formation in clouds, stepped leader, dart leader, lightning surges, switching over voltages, protection against over voltages, surge diverters, surge modifiers.

MODULE-II: DIELECTRIC BREAKDOWN (9)

Breakdown of dielectrics: Gaseous breakdown in uniform and non-uniform fields, corona discharges, breakdown of vacuum, conduction and breakdown in pure and commercial liquids, maintenance of oil quality, breakdown mechanisms in solid and composite dielectrics.

MODULE-III: GENERATION OF HIGH VOLTAGES AND HIGH CURRENTS (9)

High AC, DC voltages and currents: Generation of high DC, AC and impulse voltages and currents.

Triggering: Triggering and control of impulse generators.

MODULE-IV: MEASUREMENT OF HIGH VOLTAGES AND HIGH CURRENTS (9)

High voltage and current measurement: High resistance with series ammeter, dividers, resistance, capacitance and mixed dividers, peak voltmeter, generating voltmeters, capacitance voltage transformers, electrostatic voltmeters, sphere gaps, high current shunts, digital techniques in high voltage measurement.

MODULE-V: HIGH VOLTAGE TESTING AND INSULATION COORDINATION (9)

Testing: High voltage testing of electrical power apparatus as per international and Indian standards, power frequency, impulse voltage and dc testing of insulators, circuit breakers, bushings, isolators and transformers, insulation coordination.

IV. TEXTBOOKS:

1. S Naidu, V Kamaraju, "High Voltage Engineering", Tata McGraw-Hill, 5th Edition, 2013.
2. E Kuffel, W S Zaengl, J Kuffel, "High voltage Engineering fundamentals", Newnes, Elsevier, New Delhi, 2nd

Edition, 2005.

V. REFERENCEBOOKS:

1. L L Alston, “High Voltage Technology”, Oxford University Press, 1st Indian Edition, 2011.
2. C L Wadhwa, “High Voltage Engineering”, New Age International Publishers, 3rd Edition, 2010.
3. Subir Ray, “An Introduction to High Voltage Engineering”, PHI Learning Private Limited, New Delhi, 2nd Edition, 2013.

VI. WEBREFERENCES:

1. https://www.nptel.ac.in/courses/108104048/
2. https://www.hve.iisc.ernet.in/
3. https://www.ee.iisc.ac.in/research-hve.php
4. https://www.wikipedia.org/wiki/High_voltage
5. https://www.annauniv.edu/HighVoltage/

VII.E-TEXTBOOKS:

1. https://www.docs.google.com/file/d/0B5vXY4-Kg5GeQi1LcEU2UnJNbE0/edit
2. https://www.7see.blogspot.in/2015/04/high-voltage-engineering-by-wadhwa-free.html
3. https://www.itebooks.zone/1849192634.html
4. https://www.studynama.com/community/threads/329-High-voltage-engineering-ebook-pdf-lecture-notes-download-for-electrical

ENERGY STORAGE SYSTEMS

VIII Semester: EEE								
Course Code	Category	Hours / Week			Credits	Maximum Marks		
AEEC48	Core	L	T	P	C	CIA	SEE	Total
		3	0	0	3	30	70	100
Contact Classes:45	Tutorial Classes:Nil	Practical Classes:Nil				Total Classes:45		
Prerequisite: Chemistry, Electrical Power Transmission Systems								

I. COURSEOVERVIEW:

To enable the student to understand the emerging need for electrical energy storage, different energy storage systems and technologies available and their applications.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The characteristics of energy from various sources and need for storage.
II. The various types of energy storage and various devices used for the purpose.
III. The Identify various real time applications.

III. COURSESYLLABUS:

MODULE-I:NEEDS FOR ELECTRICAL ENERGY STORAGE (08)

Emerging needs for EES, more renewable energy, less fossil fuel, Smart Grid uses, The roles of electrical energy storage technologies, The rolesfrom theviewpointof a utility, The rolesfrom the viewpointof consumers, The rolesfromthe viewpointofgeneratorsofrenewable energy.

MODULE-II: ELECTRICAL ENERGY STORAGE TECHNOLOGIES(10)

Characteristics of electricity, Electricity and theroles of EES, High generation cost during peak-demand periods, Need for continuous andflexible supply, long distance between generation and consumption, Congestion in powergrids,Transmissionbycable.

MODULE-III: FEATURES OF ENERGY STORAGE SYSTEMS (09)

Classification of EES systems, Mechanical storagesystems,Pumped hydro storage (PHS), Compressed air energy storage (CAES), Flywheelenergy storage (FES), Electrochemical storage systems.

Secondary batteries, Flow batteries,Chemicalenergystorage,Hydrogen(H2), Synthetic naturalgas (SNG).

MODULE-IV: ELECTICAL ENERGY STORAGE SYSTEMS(10)

Electrical storage systems, Double-layer capacitors (DLC), Super conducting magnetic energy storage (SMES), Thermal storage systems, Standards for EES, Technical comparison of EES technologies.

MODULE-V: APPLICATIONS(08)

Present status of applications, Utility use (conventional power generation, gridoperation & service) , Consumer use (uninterruptable power supply for large consumers),New trends in applications ,Renewable energy generation, Smart Grid, Smart Micro grid, Smart House, Electric vehicles, Management and control hierarchy of storage systems, Internal configuration of battery storagesystems,External connection of EESsystems,Aggregating EES systems and distributed generation (Virtual Power Plant), Battery SCADA–aggregationofmanydispersedbatteries.

IV. TEXTBOOKS:

1. “James M.Eyer, Joseph J.Iannucci and Garth P.Corey, “Energy Storage Benefits and Market Analysis”, Sandia National Laboratories, 2004.

V. REFERENCEBOOKS:

1. JimEyer,GarthCorey, “Energy Storage for the Electricity Grid: Benefits and Market Potential Assessment Guide, Report, Sandia National Laboratories, Feb 2010.

VI. WEBREFERENCES:

1. https://www.researchgate.net
2. https://www.electrical4u.com

VII. E-TEXTBOOKS:

1. https://www.jntubook.com/
2. https://www.freeengineeringbooks.com

POWER QUALITY AND FACTS

VIII Semester: EEE								
CourseCode	Category	Hours/Week			Credits	Maximum Marks		
AEEC49	Elective	L	T	P	C	CIA	SEE	Total
		3	0	0	3	30	70	100
Contact Classes:45	Tutorial Classes:Nil	PracticalClasses:Nil				TotalClasses:45		
Prerequisite: Power Electronics, Power System Analysis								

I. COURSEOVERVIEW:

This course deals with the basic concepts power quality problems in distribution system, mitigation techniques for compensating devices to improve power quality in distribution system. It deals with the principle, operation and applications in power systems with respect to active/reactive power control. This course also concludes with applications of FACTS devices.

II. COURSEOBJECTIVES:

Thestudentswilltryto learn:

I. The Gain knowledge on various sources of power quality disturbances, power quality issues.
II. The Standards, measuring equipment and power quality enhancement devices.
III. The voltage sag, harmonic distortion due to commercial and industrial loads.
IV. The Design a suitable harmonic filter for industrial application.
V. The suitable custom power devices for enhancement of power quality

III. COURSESYLLABUS:

MODULE-I:POWER QUALITY ISSUES IN DISTRIBUTION SYSTEMS (08)

Power Quality problems in distribution systems: Transient and Steady state variations in voltage and frequency, unbalance, sags, swells, interruptions, wave-form distortions: harmonics, noise, notching, dc offsets, fluctuations, flicker and its measurement, Tolerance of Equipment: CBEMAcurve.

MODULE-II: CUSTOM POWER DEVICES (10)

Dynamic Voltage Restorer: Working Principle and control strategies, harmonics and unbalance mitigation in distribution systems using DSTATCOM and shunt active filters, unified power quality conditioner (UPQC), working principle, capabilities and control strategies.

MODULE-III:FACTS CONCEPTS (09)

Introduction to FACTS, BasicsofACtransmission, flow of power in AC parallel paths, meshed systems, basic types of FACTS controllers, definitions of FACTS controllers, brief description of FACTS controllers.

MODULE-IV:STATIC SHUNT AND SERIES COMPENSATORS (10)

Shunt compensation-objectives of shunt compensation, methods of controllable VAR generation, static VAR compensators- SVC, STATCOM, SVC and STATCOM comparison.

Series compensation-objectives of series compensation, thyristor switched series capacitors (TCSC), static series synchronous compensator (SSSC), power angle characteristics, and basic operating control schemes.

MODULE-V:APPLICATION OF FACTS DEVICES(08)

Application of FACTS devices for power-flow control and stability improvement. Simulation example of power swing damping in a single-machine infinite bus system using a TCSC. Simulation example of voltage regulation of transmission mid-point voltage using a STATCOM.

IV. TEXTBOOKS:

1. Narain G. Hingorani, Laszi Gyugyi, “Understanding FACTS: Concepts and Technology of Flexible AC Transmission Systems”, IEEE press, Delhi, 2nd Edition, 2001.
2. Roger C Dugan, Mark F McGranaghan, Surya Santoso, H.Wayne Beaty, “Electrical Power Systems Quality, TMH Education Pvt. Ltd., 3rd Edition, 2012.

V. REFERENCEBOOKS:

1. Mat H J Bollen, "Understanding Power Quality Problems", IEEE Press,2nd Edition,2007.
2. Arindam Ghosh, Gerard Ledwich, "Power Quality Enhancement using Custom Power Devices", Kluwer Academic Publishers, 2nd Edition, 2002.
3. Mohan Mathur, RajivKVarma,"Thyristor-based FACTS Controllers for Electrical Transmission Systems", AJohn Wiley and Sons Publications, 2nd Edition,2002.

VI. WEBREFERENCES:

1. http://www.powerqualityworld.com/.
2. https://www.researchgate.net.
3. https://www.aar.faculty.asu.edu/classes.
4. https://www.iare.ac.in

VII. E-TEXTBOOKS:

1. ArindamGhosh,GerardLedwich,Powerqualityenhancementusingcustompowerdevices,Kluweracademicpublishers , 2002.
2. https://www.freebookcentre.net.

SWITCH MODE POWER SUPPLIES

VIII Semester: EEE

Course Code	Category	Hours / Week			Credits	Maximum Marks		
AEEC50	Elective	L	T	P	C	CIA	SEE	Total
		3		-	3	40	60	100
Contact Classes: 45	Tutorial Classes: Nil	Practical Classes: Nil				Total Classes: 60		

Prerequisites: Power Electronics

I. COURSE OVERVIEW:

The power supply is a critical component in any system and has a major impact on overall reliability. System designers need to understand the designs and requirements of their power supplies to meet the evolving needs of the system, and to satisfy regulatory requirements for energy efficiency and standby power. This course covers both analog and digital switch-mode power supplies. This course starts by introducing the fundamental concepts of a real switch-mode power supply and its functions, operations and interactions. Discussions will cover the various topologies as they relate to power supply operation, design, component selection, and rating for a particular application. The course focuses on the most popular topology: the Flyback converter. The case study includes the clamp, snubber networks, transformer, and EMI filter.

II. COURSE OBJECTIVES:

Thestudentswilltryto learn:

I. Understand the concept of switched mode power supply with both D.C. and A.C. outputs.
II. Elaborately study the working of switched mode topologies including resonant power suppliers.
III. have the knowledge of their importance and applications in various fields.

III. COURSE SYLLABUS:

MODULE – I: SWITCH MODE POWER CONVERSION (09)

Introduction to Switched Mode Power Supply, Linear DC to DC Power converters, non-Idealities in reactive elements, Design of Inductors, Design of Transformers- Copper loss, Power factor, non-isolated topologies, Isolated topologies, Quasi-resonant zero-current/zero-voltage switch Operating principle of No Isolated DC to DC power Converters (Buck, Boost, Buck-Boost, and Cuk) Equivalent circuit model of the non-isolated DC-DC converters. Isolated converters (forward, Flyback).

MODULE – II: MULTIPLE OUTPUT FLYBACK SWITCH MODE POWER SUPPLIES(09)

Introduction, operating Modes, operating principles, Direct off line Flyback Switch Mode Power Supplies, Flyback converter, snubber network, Problems.

MODULE – III: USING POWER SEMICONDUCTORS IN SWITCHED MODE TOPOLOGIES (09)

Introduction to Switched Mode Power Supply Topologies, The Power Supply Designer's Guide to High Voltage Transistors, Base Circuit Design for High Voltage Bipolar Transistors in Power Converters, Isolated Power Semiconductors for High Frequency Power Supply Applications.

MODULE -IV: RECTIFICATION (09)

Explanation, Advantages and disadvantages, SMPS and linear power supply comparison, Theory of operation , Input rectifier stage, Inverter stage, Voltage converter. and output rectifier, Regulation, An Introduction to Synchronous Rectifier Circuits using Power MOS Transistors.

MODULE – V: SWITCH MODE VARIABLE POWER SUPPLIES (09)

Introduction, variable SMPS techniques, operating principles, practical limiting factors, Efficiency and EMI Applications. Resonant Power Supplies: An Introduction to Resonant Power Supplies, Resonant Power Supply Converters - The Solution for Mains Pollution Problems.

IV. TEXT BOOKS:

1. "Keith H. Billings and Taylor Morey", "Switch Mode Power Supplies", Tata McGraw-Hill Publishing Company, 3rd Edition 2010.
2. "Robert W. Erickson", "Switch Mode Power Supplies", Springer, 2nd Edition 2001.

V. REFERENCE BOOKS:

1. Sanjaya Maniktala, “Switching Power Supplies A-Z”, Elsevier, 2nd Edition, 2012.
2. Steven M. Sandler, “Switch Mode Power Supplies”, Tata McGraw Hill, 1st Edition, 2006.

VI. WEB REFERENCES:

1. https://www.eeweb.com
2. https://www.sciencedirect.com
3. https://www.onsemi.com
4. https://www.iare.ac.in

VII. E-Text Books:

1. https://www.kobo.com
2. https://www.accessengineeringlibrary.com
3. https://www.alibris.com
4. https://www.amazon.in

UTILIZATION OF ELECTRICAL POWER

VIII Semester: EEE									
CourseCode	Category	Hours/Week			Credits	Maximum Marks		
AEEC51	Core	L	T	P	C	CIA	SEE	Total
		3	0	0	3	30	70	100
Contact Classes:45	Tutorial Classes:Nil	PracticalClasses:Nil				TotalClasses:45		
Prerequisite: DC Machine and Transformers, AC Machines								

I. COURSE OVERVIEW:
This course deals with effective utilization of electrical power. It describes the basic principles and performa characteristics of drives, electric traction and illumination, it also facilitates the use of electrical power for domestic industrial consumers.

II. COURSE OBJECTIVES:
Thestudentswilltryto learn:

I. The performance characteristics of electrical drives and their deployment with different loading environment.
II. The importance of electrical power in various utilities with illumination, heating and welding.
III. The impact of acceleration, braking retardation and adhesive weight in electric traction system.

III. COURSE SYLLABUS:
MODULE-I:ELECTRIC DRIVES (09)

Type of electric drives, choice of motor, starting and running characteristics, speed control, particular applications of electric drives, types of industrial loads, continuous, intermittent and variable loads, load equalization.

MODULE-II:ELECTRIC HEATING AND WELDING (09)

Electric heating: Advantages and methods of electric heating, resistance heating induction heating and dielectric heating: Electric welding: resistance and arc welding, electric welding equipment, comparison between AC and Welding.

MODULE-III:ILLUMINATION (09)

Illumination: Introduction, terms used in illumination, laws of illumination, polar curves, photometry, integrating sphere.

Sources of light: Discharge lamps, MV and SV lamps, comparison between tungsten filament lamps and fluorescent tubes, basic principles of light control, types and design of lighting and flood lighting.

MODULE-IV:TRAIN MECHANICS (09)

System of electric traction and track electrification, review of existing electric traction systems in India, special features of traction motor, methods of electric braking-plugging, rheostat braking and regenerative braking, mechanics of train movement, speed-time curves for different service: Trapezoidal and quadrilateral speed time curves.

MODULE-V:ELECTRIC TRACTION (08)

Calculations of tractive effort, power, specific energy consumption for given run, effect of varying acceleration and braking retardation, adhesive weight and braking retardation adhesive weight and coefficient of adhesion.

II. TEXT BOOKS:

1. S Sivarnagaraju, D Srilatha, M Balasubbareddy, “Generation and Utilization of Electrical Energy”, Pearson Education India, 1st Edition, 2010.
2. E Openshaw Taylor, Orient Longman, “Utilizations of Electric Energy”, 1st Edition, 2003.

V. REFERENCEBOOKS:

1. N V Suryanarayana, “Utilization of Electrical Power including Electric drives and Electric traction New Age International (P) Limited, Publishers, 1st Edition, 1996.
2. C L Wadhwa, “Generation, Distribution and Utilization of electrical Energy”, New Age International (P)

Limited, 1st Edition, 1997.
3. Partab, "Art & Science of Utilization of Electrical Energy", Dhanpat Rai & Sons, 2nd Edition, 2000.

VI. WEBREFERENCES:

1. https://www.researchgate.net
2. https://www.aar.faculty.asu.edu/classes
3. https://www.facstaff.bucknell.edu/
4. https://www.electrical4u.com
5. https://www.iare.ac.in

VII. E-TEXTBOOKS:

1. https://www.jntubook.com/
2. https://www.freeengineeringbooks.com
3. https://www.freevideolectures.com

INDUSTRIAL ELECTRICAL SYSTEMS

VIII Semester: EEE

CourseCode	Category	Hours/Week			Credits	Maximum Marks		
AEEC52	Core	L	T	P	C	CIA	SEE	Total
		3	0	0	3	30	70	100
Contact Classes:45	Tutorial Classes: Nil	Practical Classes: Nil				TotalClasses:45		

Prerequisite: DC Machine and Transformers

I. I. COURSEOVERVIEW:

Overview of Electrical Systems in Manufacturing, Chemical, Metallurgical, Process Industries, Electric Traction, Electric Heating, Electric Welding, Electroplating, Illumination and case studies.

II. COURSE OBJECTIVES:

The students will try to learn:

I. the proper size of various electrical system components.
II. the electrical wiring systems for residential, commercial and industrial.
III. the basic quantities of light, definitions and relationships kinds of lamps, characteristics and Lighting calculations and illumination technology
IV. The various components of industrial electrical systems with automation.

III. COURSESYLLABUS:

MODULE – I: ELECTRICAL SYSTEM COMPONENTS (08)

Electrical System Components: Introduction to LT and HT system, Contactor, Isolator, Relays, metering system, Tariff structure, inverse current characteristics, components symbol, single line diagram (SLD) of a wiring system. Protection components: Fuse, MCB, MCCB, ELCB, MPCB.

MODULE – II: RESIDENTIAL AND COMMERCIAL ELECTRICAL SYSTEMS (09)

Residential and Commercial Electrical Systems: electric supply system, three phase four wire distribution system, protection of electric installation against over load, short circuit and earth fault, earthing, guide lines for installation of fittings, general requirements of electrical installations, testing of installations.

MODULE – III: ILLUMINATION SYSTEMS (10)

Illumination Systems: Production of light, Laws of illumination, lighting calculation, Interior and exterior illumination systems, lighting schemes, design on lighting scheme;

Electrical lamps, factory lighting, flood lighting, gaseous discharge lamps, high pressure and low-pressure neon lamps, high frequency, low pressure discharge tubes, induction lamps, LED lamps, Simple problems.

MODULE -IV: INDUSTRIAL ELECTRICAL SYSTEMS (9)

Industrial Electrical Systems: Indian electricity rules, neutral and earth wire, types of loads, systems of wiring, service connections, service mains, sub-circuits, location of outlets; location of control switches, location of main board and distribution board, load assessment, permissible voltage drops and sizes of wires, estimating and costing of electric installations. Types of DG systems and UPS system, battery selection, types of battery and battery bank.

MODULE – V: INDUSTRIAL ELECTRICAL AUTOMATION SYSTEMS (9)

Industrial Electrical Automation Systems: introduction to basic PLC, Role of in automation, advantages of process automation, PLC based control system design, Panel Metering and Introduction to SCADA system for distribution automation.

IV. **TEXT BOOKS:**

1. S L Uppal and G C Garg, "Electrical Wiring, Estimating & Costing", Khanna publishers,2nd Edition, 2008.
2. K B Raina, "Electrical Design, Estimating & Costing", New age International, 2nd Edition, 2007.
3. S Singh and R D Singh, "Electrical estimating and costing", Dhanpat Rai and Co., 1st Edition, 1997.

V. REFERENCEBOOKS:

1. Amman Jordan, Mohammad TawfeeqALZu "hairi, Electrical Installation for buildings & facilities", 2nd Edition 2016.
2. H. Joshi, "Residential Commercial and Industrial Systems", McGraw Hill Education, 3rd Edition, 2008.
3. Gupta J B Katson, Ludhiana, "Electrical Installation, estimating and costing", S K Kataria and sons, 3rdEdition, 2013.

VI. WEBREFERENCES:

1. http://www.turfproductscorp.com/wp-content/uploads/2017/07/Service
2. https://en.wikipedia.org/wiki/Electric_power_system
3. https://nptel.ac.in/courses/Webcourse

VII. E-TEXTBOOKS:

1. http://www.newelectric.com/what-we-do/services/industrial-automation/
2. https://en.wikipedia.org/wiki/Electric_light

RENEWABLE ENERGY AND SMART GRID

VIII Semester: EEE

Course Code	Category	Hours / Week			Credits	Maximum Marks		
AEEC53	Elective	L	T	P	C	CIA	SEE	Total
		3		-	3	40	60	100
Contact Classes: 45	Tutorial Classes: Nil	Practical Classes: Nil				Total Classes: 60		

Prerequisites: Power Electronics in Renewable Energy Systems

I. COURSE OVERVIEW:

The course introduces students to the history of the Indian power grid and to the basic concepts of the current electric power system. The main challenges of the transition from the traditional power system with unidirectional power flow to the new and complex system connected to renewable sources and bidirectional power flow capability is also presented in this course. In addition, the impact of distributed generation and electric vehicles is discussed along with cybersecurity and information privacy issues inherent in this new power grid.

II. COURSE OBJECTIVES:

The students will try tolearn:

I. The concepts, architecture and design of smart grids.
II. The communication and measurement technologies employed in smart grid.
III. The tools for the performance analysis and stability analysis of smart grid.
IV. The renewable energy resources and storages integrated with smart grid.

III. COURSE SYLLABUS:

MODULE – I: INTRODUCTION(9)

Today's Gird Versus Smart Grid, Rationale for Smart Grid, Computational Intelligence, Power System Enhancement, Communication and Standards, Environment and Economics, Shareholders Roles and Function, Architecture, Functions of Components.

MODULE – II: SENSORS and MEASUREMENT (9)

Sensors for Smart Grid, Monitoring and Measurement Technologies, PMU, Smart meters, Smart Appliances, Multi Agent Systems (MAS) Technology, Micro grid and Smart grid comparison, Wide Area Measurement

MODULE – III: USING POWER SEMICONDUCTORS IN SWITCHED MODE TOPOLOGIES (9)

Introduction to Switched Mode Power Supply Topologies, The Power Supply Designer's Guide to High Voltage Transistors;

Base Circuit Design for High Voltage Bipolar Transistors in Power Converters, Isolated Power Semiconductors for High Frequency Power Supply Applications.

MODULE -IV: ENERGY STORAGE (9)

Batteries, Flow Batteries, Fuel Cell and hydrogen electrolytes, Flywheel, Super conduction magnetic energy storage systems, super capacitors, Simulation and case studies.

MODULE – V: ELECTRICAL VEHICLES (9)

Plugin Electric Vehicles and hybrid, Vehicle classes, Vehicle Architecture, Gird to Vehicle (G2V) Charging, Grid Impacts, Vehicle to Grid (V2G), cybersecurity.

IV. TEXT BOOKS:

1. James Momoh, "Smart Grid: Fundamentals of design and analysis", John Wiley & sons Inc, IEEE press 2012.
2. Janaka Ekanayake, Nick Jenkins, Kithsiri Liyanage, Jianzhong Wu, Akihiko
3. Yokoyama, "Smart Grid: Technology and Applications", John Wiley & Sons Inc, 2012.
4. Lars.T.Berger, K.Iniewski, "Smart Grid: Applications, Communications & Security" Wiley India Pvt. Ltd, Reprint 2015.

V. REFERENCE BOOKS:

1. Fereidoon P. Sioshansi, “Smart Grid: Integrating Renewable, Distributed & Efficient Energy”, Academic Press, 2012.
2. Clark W.Gellings, “The smart grid: Enabling energy efficiency and demand response”, Fairmont Press Inc, 2009.
3. Qi Huang, Shi Jing “Innovative Testing and Measurement Solutions for Smart Grid”, John Wiley & Sons Inc, 2015.

VI. Web References:

1. https://www.electricalindia.in
2. https://www.energy.gov
3. https://www.whatis.techtarget.com
4. https://www.iare.ac.in

VII.E-Text Books:

1. https://www.e4uhu.com
2. https://www.onlinelibrary.wiley.com
3. https://www.smartgrid.gov
4. https://www.amazon.in

ELECTRICAL AND HYBRID VEHICLES

VIII Semester: EEE								
Course Code	Category	Hours / Week			Credits	Maximum Marks		
AEEC54	Elective	L	T	P	C	CIA	SEE	Total
		3		-	3	40	60	100
Contact Classes: 45	Tutorial Classes: Nil	Practical Classes: Nil				Total Classes: 60		
Prerequisites: DC Machine and Transformers, AC Machines								

I. COURSE OVERVIEW:

This course introduces the fundamental concepts, principles, analysis and design of hybrid and electric vehicles. This course goes deeper into the various aspects of hybrid and electric drive train such as their configuration, types of electric machines that can be used, energy storage devices, etc. Each topic will be developed in logical progression with up-to-date information.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The working principle of electric vehicles.
II. The hybrid vehicle configuration and its components, performance analysis.
III. The construction and working principle of various motors used in electric vehicles.
IV. Choose proper energy storage systems for vehicle applications.
V. The various communication protocols and technologies used in vehicle networks.

III. COURSE SYLLABUS:

MODULE-I: INTRODUCTION TO HYBRID ELECTRIC VEHICLES

History of hybrid and electric vehicles, social and environmental importance of hybrid and electric vehicles, impact of modern drive-trains on energy supplies. Conventional Vehicles: Basics of vehicle performance, vehicle power source characterization, transmission characteristics, mathematical models to describe vehicle performance.

MODULE-II: HYBRID ELECTRIC DRIVE-TRAINS

Basic concept of hybrid traction, introduction to various hybrid drive-train topologies, power flow control in hybrid drive-train topologies, fuel efficiency analysis. Electric Drive-trains: Basic concept of electric traction, introduction to various electric drive-train topologies, power flow control in electric drive-train topologies, fuel efficiency analysis.

MODULE-III: ELECTRIC PROPULSION UNIT

Introduction to electric components used in hybrid and electric vehicles, Configuration and control of DC Motor drives, Configuration and control of Induction Motor drives.

Configuration and control of Permanent Magnet Motor drives, Configuration and control of Switch Reluctance Motor drives, drive system efficiency.

MODULE-IV: ENERGY STORAGE

Introduction to Energy Storage Requirements in Hybrid and Electric Vehicles, Battery based energy storage and its analysis, Fuel Cell based energy storage and its analysis, Super Capacitor based energy storage and its analysis, Flywheel based energy storage and its analysis, Hybridization of different energy storage devices.

MODULE-V: SIZING THE DRIVE SYSTEM

Matching the electric machine and the internal combustion engine (ICE), Sizing the propulsion motor, sizing the power electronics, selecting the energy storage technology, Communications, supporting subsystems.

VI. TEXT BOOKS

1. Iqbal Hussein, Electric and Hybrid Vehicles: Design Fundamentals, II Edition, CRC Press, 2011
2. James Larminie, “Electric Vehicle Technology Explained”, John Wiley & Sons, 2003.

VII. REFERENCEBOOKS:

1. Mehrdad Ehsani, Yimin Gao, Ali Emadi, “Modern Electric, Hybrid Electric, and Fuel Cell Vehicles: Fundamentals”, CRC Press, 2010.
2. Sandeep Dhameja, “Electric Vehicle Battery Systems”, Newnes, 2000.
3. Modern Electric, Hybrid Electric, and Fuel Cell Vehicles, Mehrdad EhsaniYimin Gao Stefano Longo Kambiz M. Ebrahimi, Taylor & Francis Group, LLC, 2018.

VIII. WEBREFERENCES:

1. https://www.researchgate.net
2. https://www.aar.faculty.asu.edu/classes
3. https://www.facstaff.bucknell.edu/
4. https://www.electrical4u.com
5. https://www.iare.ac.in

IX. E-TEXTBOOKS:

1. https://www.springer.com
2. https://www.ieeexplore.ieee.org

FLIGHT CONTROL THEORY

OE –I: VI Semester: AERO / MECH / CIVIL OE – III: VIII Semester: CSE / CSE (AI & ML) / CSE (DS) / CSE (CS) / CSIT / IT / ECE / EEE								
Course Code	Category	Hours / Week			Credits	Maximum Marks		
AAEC30	Elective	L	T	P	C	CIA	SEE	Total
		3	-	-	3	30	70	100
Contact Classes: 45	Tutorial Classes: Nil	Practical Classes: Nil				Total Classes: 45		

I. COURSE OVERVIEW:

Flight control system of an aircraft is instrumental in establishing stability of the aircraft through control surfaces. This course introduces the concepts of the control system theory such as transfer functions, step response and impulse response. This course covers stability, feedback and different techniques used for control systems analysis. The course emphasizes on the flight control systems, response analysis for control surface inputs and control augmentation systems such as autopilots.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The stability criteria to determine the stability of an aircraft, and specify the aircraft time-domain and frequency-domain response specifications.
II. The classical control theory in the frequency domain and modern control theory in the state- space are effectively mixed to provide the student with a modern view of systems theory.
III. The various control techniques for aircraft control systems, and study some feedback control applications.
IV. The controllability and observability of aerospace systems, and apply the modern control techniques to design enhanced flight control systems.

III. COURSE SYLLABUS:

MODULE-I: INTRODUCTION TO CONTROL SYSTEM (10)

Dynamical systems-principal constituents-input, output-process (plant)-block diagram representation. Inputs- control input, noise. Function of controls regulation (hold), tracking (command)-examples. Measure of effectiveness. Sensitivity of output to control input, noise and system parameters-robustness. Deterministic and stochastic control. Control in everyday life. The pervasiveness of control in nature, engineering and societal systems. The importance of study of control system. Need for stable, effective (responsive), robust control system. Modeling of dynamical systems by differential equations-system parameters. Examples from diverse fields. First and second order systems, higher order systems, single input single output systems, and multiple-input multiple-output.

MODULE –II: MATHEMATICAL MODELING OF DYNAMICAL SYSTEMS (10)

Control system performance- time domain description- output response to control inputs-- impulse and indicial response- characteristic parameters- significance- relation to system parameters- examples- first and second order linear systems, higher order systems. Synthesis of response to arbitrary input functions from impulse and indicial response. Review of Fourier transforms and Laplace transforms-inverse transforms- significance, applications to differential equations. 's' (Laplace) domain description of input- output relations- transfer function representation- system parameters- gain, poles and zeroes. Characteristic equation- significance- examples. Frequency and damping ratio of dominant poles. Relation of transfer functions to impulse response. Partial fraction decomposition of transfer functions-significance.

MODULE –III: STEADY STATE RESPONSE ANALYSIS (10)

System type, steady state error, error constants- overall system stability. Application of feedback in stability augmentation, control augmentation, automatic control-examples. Composition, reduction of block diagrams of complex systems-rules and conventions. Control system components - sensors, transducers, servomotors, actuators, filters-modeling, transfer functions. Single-input single-output systems. Multiple input-multiple output systems, matrix transfer functions-examples. Types of control problems- the problem of analysis, control synthesis, system synthesis- examples- static control of aircraft.

Extension to dynamic control. System identification from input output measurements importance. Flight path stabilization, longitudinal control law design using back stepping algorithm. Experimental determination of system transfer functions by frequency response measurements. Example. Frequency domain description- frequency response- gain and phase shift- significance- representation asymptotic (Bode) plots, polar (Nyquist) plots, frequency transfer functions. Characteristic parameters corner frequencies, resonant frequencies, peak gain, and bandwidth- significance. First and second order systems- extension to higher order systems.

MODULE –IV: AIRCRAFT RESPONSE TO CONTROL (07)

Approximations to aircraft transfer functions, control surface actuators-review. Response of aircraft to elevator input, Response of aircraft to rudder input and Response of aircraft to aileron input to atmosphere. Need for automatic control. Auto pilots Stability augmentation systems-pitch damper and yaw damper.

MODULE –V: FLYING QUALITIES OF AIRCRAFT (08)

Reversible and irreversible flight control systems. Flying qualities of aircraft-relation to airframe transfer function. Pilot's opinion ratings. Flying quality requirements- pole-zero, frequency response and time- response specifications. Displacement and rate feedback determination of gains conflict with pilot input s resolution-control augmentation systems- Full authority fly-by-wire. Auto Pilot-Normal acceleration, Turn rate, Pitch rate Commands-Applications.

IV. TEXT BOOKS:

1. Kuo, B.C., “Automatic control of Aircraft and Missiles”, John Wiley Sons, New York, 1990.
2. Stevens B.L & Lewis F.L, “Aircraft control & Simulation”, John Wiley Sons, New York, 1992.

V. REFERENCE BOOKS:

1. Mc Lean, D., “Automatic Flight Control Systems”, Prentice Hall, 1990.
2. Bryson, A.E., “Control of Aircraft and Spacecraft”, Princeton University Press, 1994.
3. E H J Pallett, Shawn Coyle, “Automatic Flight Control”, 4th Edition, 2002.

VI. WEB REFERENCES:

1. https://www.e-booksdirectory.com/
2. https://www.aerospaceengineering.es/book/

VII. E-TEXT BOOKS:

1. https://books.google.co.in/books?isbn=1118870972
2. https://books.google.co.in/books?isbn=0387007261

AIRFRAME STRUCTURAL DESIGN

OE –I: VI Semester: AERO / MECH / CIVIL **OE – III:** VIII Semester: CSE / CSE (AI & ML) / CSE (DS) / CSE (CS) / CSIT / IT / ECE / EEE								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
AAEC31	**Elective**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		3	-	-	3	30	70	100
Contact Classes: 45	**Tutorial Classes: Nil**	**Practical Classes: Nil**				**Total Classes: 45**		

I. COURSE OVERVIEW:

This course deals with fundamental aspects of an anatomy of aircraft and the current trends in airframe design. It includes the evolution of the aircraft and space industry, aerodynamics and performance of the aircraft with their applications. It compares and contrasts various thrust vector control mechanisms of different aircraft propulsion systems. It discusses various materials and its properties that are used for manufacturing different parts of an aircraft. This course enriches the knowledge of connection between theoretical and practical methods for performing the airframe design exercises

II. COURSE OBJECTIVES:

The student will try to learn:

I. The fundamental concepts of various airframe designs, aircraft propulsion systems and aerodynamic forces/moments acting on the aircraft and spacecraft under static and dynamic load conditions

II. The characteristics of stability and performance of an aircraft and the role of primary and secondary controls in longitudinal and lateral stability

III. The properties of different materials that are used in industries for manufacturing various components of an aircraft and spacecraft achieving specified stability requirements.

IV. The mathematical modeling of tailless aircraft, flapping wing aircraft and innovative designs in modern aircraft for future requirements.

II. COURSE SYLLABUS:

MODULE-I: HISTORY OF FLIGHT AND SPACE ENVIRONMENT (10)

Balloons and dirigibles, heavier than air aircraft, commercial air transport; Introduction of jet aircraft, helicopters, missiles; Conquest of space, commercial use of space; Different types of flight vehicles, classifications exploring solar system and beyond, a permanent presence of humans in space; Earth‘s atmosphere, the standard atmosphere; The temperature extremes of space, laws of gravitation, low earth orbit, microgravity, benefits of microgravity; Environmental impact on spacecraft, space debris; Planetary environments.

MODULE –II: INTRODUCTION TO AERODYNAMICS (10)

Anatomy of the airplane, helicopter; Understanding engineering models; Aerodynamic forces on a wing, force coefficients; Generating lift, moment coefficients; Aerodynamic forces on aircraft – classification of NACA airfoils, aspect ratio, wing loading, mach number, centre of pressure and aerodynamic centre, aerofoil characteristics-lift, drag curves; Different types of drag.

MODULE –III: FLIGHT VEHICLE PERFORMANCE AND STABILITY (09)

Performance parameters, performance in steady flight, cruise, climb, range, endurance, accelerated flight symmetric maneuvers, turns, sideslips, takeoff and landing.

Flight vehicle Stability, static stability, dynamic stability; Longitudinal and lateral stability; Handling qualities of the airplanes.

MODULE –IV: INTRODUCTION TO AIRPLANE STRUCTURES AND MATERIAL,POWERPLANT (08)

General types of construction, monocoque, semi-monocoque; Typical wing and fuselage structure; Metallic & non-metallic materials, use of aluminum alloy, titanium, stainless steel and composite materials; Basic ideas about engines, use of propellers and jets for thrust production; Principles of operation of rocket, types of rockets.

MODULE –V: SATELLITE SYSTEMS ENGINEERING HUMAN SPACE EXPLORATION (08)

Satellite missions, an operational satellite system, elements of satellite, satellite bus subsystems; Satellite structures, mechanisms and materials; Power systems; Communication and telemetry; Propulsion and station keeping; Space missions, mission objectives. Goals of human space flight missions, historical background, the Soviet and US missions; The mercury, Gemini, Apollo (manned flight to the moon), Skylab, apollo-soyuz, space Shuttle; International space station, extravehicular activity; The space suit; The US and Russian designs; Life support systems, flight safety; Indian effort in aviation, missile and space technology.

IV. TEXT BOOKS:

1. Newman D, “Interactive Aerospace Engineering and Design”, McGraw-Hill, 1st Edition, 2002.
2. Anderson J. D, “Introduction To Flight”, McGraw-Hill Education, 5th Edition, 2002.

V. REFERENCE BOOKS:

1. Kermode. A. C, “Flight without Formulae”, McGraw Hill, 4th Edition, 1997.
2. Barnard R.H and Philpot. D.R, “Aircraft Flight”, Pearson, 3rd Edition, 2004.
3. SwattonP.J, “Flight Planning”, Blackwell Publisher, 6th Edition, 2002.

VI. WEB REFERENCES:

1. http://ase.sbu.ac.ir/FA/Staff/abbasrahi/Lists/Dars/Attachments/11/Vibrations%20of%20Continuous%20Systems.pdf
2. http://arc-test.aiaa.org/doi/book/10.2514/4.862458
3. http://arc-test.aiaa.org/doi/abs/10.2514/5.9781600862373.0719.0728

VII. E-TEXT BOOKS:

1. http://www.gregorypaulblog.com/structural-dynamics-in-aeronautical-engineering-aiaa-education-series.pdf
2. https://aerocastle.files.wordpress.com/2012/10/mechanical_vibrations_5th-edition_s-s-rao.pdf

INDUSTRIAL MANAGEMENT

OE –I: VI Semester: AERO / MECH / CIVIL OE – III: VIII Semester: CSE / CSE (AI & ML) / CSE (DS) / CSE (CS) / CSIT / IT / ECE / EEE							
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
AMEC34	**Elective**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		3	0	0	3	30	70	100
Contact Classes: 45	**Tutorial Classes: Nil**	**Practical Classes: Nil**				**Total Classes: 45**		

I.COURSE OVERVIEW:

The industrial management prepares engineers to design, improve, install, and operate the integrated systems of people, materials, and facilities needed by industry, commerce, and society. Industrial engineers solve problems that arise in the management of systems, applying the principles of engineering science, product/service and process design, work analysis, human factors principles, and operations research.The focus of this course is how to improve processes or design things that are more efficient and waste less money, time, raw resources, man-power and energy while following safety standards and regulations

II. COURSE OBJECTIVES:

The students will try to learn:

I. The production planning and control procedures to handle industrial disputes.
II. The Work study procedures and quality concepts to enhance more productivity
III. The significant exposure on some maintenance practices in industry for consistent productivity.

III.COURSE SYLLABUS:

MODULE-I: CONCEPTS OF INDUSTRIAL MANAGEMENT (9)

Principles of management- Growth of management thought, Functions of management, Principles of organization, Types of organization and committees.

MODULE –II: WORK STUDY (9)

Concept of productivity, Method Study - Basic steps in method study, Process charts, Diagrams, Principles of motion economy, Micro motionstudy, Therbligs, SIMO chart. Work Measurement - Stop watch procedure of time study, Performance rating, allowances, Work sampling, Simple problems.

MODULE –III: INVENTORY CONTROL (9)

Inventory Control: Inventory, Cost, Deterministic Models and Introduction to Supply Chain Management.

MODULE –IV: QUALITY CONTROL (9)

Quality Control: Process control, SQC, Control charts, Single, Double and Sequential Sampling, Introduction to TQM.

MODULE –V: DEMAND FORECASTING AND COST ESTIMATION (9)

Demand Forecasting and cost Estimation: Characteristics of Forecasts, Forecasting Horizons, Steps to Forecasting, Forecasting Methods, Seasonal Adjustments, Forecasting Performance Measures, Cost Estimation, Elements of cost, Computation of Material Variances Break-Even Analysis.

IV. TEXT BOOKS:

1. O.P. Khanna, "Industrial Engineering and Management", Khanna Publishers.
2. T.R. Banga and S.C.Sarma, "Industrial Engineering and Management Science", Khanna Publishers.

V. REFERENCE BOOKS:

1. Ralph M Barnes, "Motion and Time Study", John Willey & Sons Work Study lLO.

2. Ernest J McCormick, “Human factors in Engineering & Design”, TMH.
3. Paneer Selvam, “Production & Operation Management”, PHI.
4. NVS Raju, “Industrial Engineering Management”, Cengage Learning.

VI. REFERENCE BOOKS:

1. https://nptel.ac.in/courses/112/107/112107142/#
2. https://nptel.ac.in/courses/112/107/112107143/#

ELEMENTS OF MECHANICAL ENGINEERING

<table>
<tr><td colspan="9">OE –I: VI Semester: AERO / MECH / CIVIL
OE – III: VIII Semester: CSE / CSE (AI & ML) / CSE (DS) / CSE (CS) / CSIT / IT / ECE / EEE</td></tr>
<tr><td rowspan="3">Course Code

AMEC35</td><td rowspan="3">Category

Elective</td><td colspan="3">Hours / Week</td><td>Credits</td><td colspan="3">Maximum Marks</td></tr>
<tr><td>L</td><td>T</td><td>P</td><td>C</td><td>CIA</td><td>SEE</td><td>Total</td></tr>
<tr><td>3</td><td>0</td><td>0</td><td>3</td><td>30</td><td>70</td><td>100</td></tr>
<tr><td>Contact Classes: 45</td><td>Tutorial Classes: Nil</td><td colspan="4">Practical Classes: Nil</td><td colspan="3">Total Classes: 45</td></tr>
</table>

I.COURSE OVERVIEW:

The main aim of this course to impart mechanical engineering fundamental basics to allied engineering students so that they have minimum understanding of mechanical system, equipment and process.

II. COURSE OBJECTIVES:

The students will try to learn:

I. The fundamentals of mechanical systems.
II. The significance of mechanical engineering and apply in different fields of engineering.
III. The various applications of engineering materials for designing different engineering components.

III. COURSE SYLLABUS:

MODULE-I: SOURCES OF ENERGY, BASIC CONCEPTS OF THERMODYNAMICS (9)

Sources of Energy : Introduction and application of energy sources like fossil fuels, hydel, solar, wind, nuclear fuels and bio-fuels; environmental issues like global warming and ozone depletion.

Basic concepts of Thermodynamics: Introduction, states, concept of work, heat, temperature; Zeroth, 1st, 2nd and 3rd laws of thermodynamics. Concept of internal energy, enthalpy and entropy (simple numericals).

MODULE –II: BOILER AND TURBINES(9)

Boilers: Introduction to boilers, classification, Lancashire boiler, Babcock and Wilcox boiler. Introduction to boiler mountings and accessories (no sketches).

Turbines: Hydraulic Turbines-Classification and specification, Principles and operation of Pelton wheel turbine, Francis turbine and Kaplan turbine (elementary treatment only).

Hydraulic Pumps: Introduction, classification and specification of pumps, reciprocating pump and centrifugal pump, concept of cavitations and priming.

MODULE –III: PROPERTIES, COMPOSITION AND INDUSTRIAL APPLICATIONS OF ENGINEERING MATERIALS(9)

Metals-Ferrous: cast iron, tool steels and stainless steels and nonferrous: aluminum, brass, bronze. Polymers -Thermoplastics and thermosetting polymers. Ceramics -Glass, optical fiber glass, cermets. Composites -Fiber reinforced composites, Metal Matrix Composites, Smart materials -Piezoelectric materials, shape memory alloys, semiconductors and insulators.

Joining Processes: Soldering, Brazing and Welding Definitions. Classification and methods of soldering, brazing and welding. Brief description of arc welding, oxy-acetylene welding, TIG welding, and MIG welding.

MODULE –IV: MACHINE TOOLS(9)

Lathe -Principle of working of a center lathe. Parts of a lathe. Operations on lathe –Turning, Facing, Knurling, Thread Cutting, Drilling, Taper turning by Tailstock offset method and Compound slide swiveling method, Specification of Lathe.

Milling Machine-Principle of milling, types of milling machines. Working of horizontal and vertical milling machines. Milling processes -plane milling, end milling, slot milling, angular milling, form milling, straddle milling, and gang milling.

MODULE –V: INTRODUCTION TO ADVANCED MANUFACTURING SYSTEMS (9)

Computer Numerical Control (CNC): Introduction. Components of CNC, open loop and closed loop systems, advantages of CNC, CNC Machining centers and Turning centers.

Robots: Robot anatomy, joints and links, common robot configurations. Applications of Robots in material handling, processing and assembly and inspection

TEXT BOOKS

V. K. Manglik, "Elements of Mechanical Engineering", Prentice Hall, 1st Edition, 2013.

Mikell P. Groover, "Automation, Production Systems and CIM", Prentice Hall, 4th Edition, 2013

REFERENCE BOOKS:

S. Trymbaka Murthy, "A Text Book of Elements of Mechanical Engineering", University Press, 4th Edition, 2006.

K. P. Roy, S. K. Hajra Choudary, Nirjhar Roy, "Element of Mechanical Engineering", Media Promoters & Publishers, 7th Edition, 2012.

Pravin Kumar, "Basic Mechanical Engineering", Pearson, 1st Edition, 2013

WEB REFERENCES:

http://www.nptel.ac.in/courses/112107144/

http://www.nptel.ac.in/courses/112101098/download/lecture-37.pdf

MODERN CONSTRUCTION MATERIALS

OE –I: VI Semester: AERO / MECH / CIVIL OE – III: VIII Semester: CSE / CSE (AI & ML) / CSE (DS) / CSE (CS) / CSIT / IT / ECE / EEE							
Course Code	Category	Hours / Week			Credits	Maximum Marks		
ACEC30	Elective	L	T	P	C	CIA	SEE	Total
		3	0	0	3	30	70	100
Contact Classes: 45	Total Tutorials: Nil	Total Practical Classes: Nil				Total Classes: 45		

I. COURSE OVERVIEW:

This course provides the scientific basis for the understanding and development of construction materials. It serves as a foundation course for post-graduate students interested in careers involving research, teaching and/or construction engineering, as well as marketing, decision making, innovation and specification related to construction materials.

II. COURSE OBJECTIVES:

The student will try to learn:

I. The concept of modern water proofing and insulating materials in constructions.
II. Importance of composites and chemicals in production of modern concrete.
III. The types of concrete and their constituents and properties.
IV. The impact of building construction on society and demonstrate awareness of contemporary issues.

III.COURSE SYLLABUS:

MODULE-I: STONES – BRICKS – CONCRETE BLOCKS (09)

Stone as building material, Criteria for selection, Tests on stones, Deterioration and Preservation of stone work, Bricks, Classification, Manufacturing of clay bricks, Tests on bricks Compressive Strength, Water Absorption, Efflorescence, Bricks for special use, Refractory bricks, Cement, Concrete blocks, Light-weight concrete blocks.

MODULE-II: LIME – CEMENT – AGGREGATES – MORTAR (09)

Lime, Preparation of lime mortar, Cement, Ingredients, Manufacturing process, Types and Grades, Properties of cement and Cement mortar, Hydration, Compressive strength, Tensile strength, Fineness, Soundness and consistency, Setting time, Industrial byproducts, Fly ash, Aggregates, Natural stone aggregates, Crushing strength, Impact strength, Flakiness Index, Elongation Index, Abrasion Resistance, Grading, Sand Bulking.

MODULE-III: CONCRETE (09)

Concrete, Ingredients, Manufacturing Process, Batching plants, RMC, Properties of fresh concrete, Slump, Flow and compaction Factor, Properties of hardened concrete, Compressive, Tensile and shear strength.

Modulus of rupture, Tests, Mix specification, Mix proportioning, BIS method, High Strength Concrete and HPC, Self compacting Concrete, Other types of Concrete, Durability of Concrete.

MODULE-IV: TIMBER AND OTHER MATERIALS (09)

Timber, Market forms, Industrial timber, Plywood, Veneer, Thermacole, Panels of Laminates, Steel, Aluminum and Other Metallic Materials, Composition, Aluminium composite panel, Uses: Market forms, Mechanical treatment, Paints, Varnishes, Distempers, Bitumens.

MODULE-V: MODERN MATERIALS (09)

Glass, Ceramics, Sealants for joints, Fibre glass reinforced plastic, Clay products, Refractories, Composite materials, Types, Applications of laminar composites, Fibre textiles, Geomembranes and Geotextiles for earth reinforcement.

IV.TEXT BOOKS:

1. W.D. Callister, John Wiley, “Materials Science and Engineering: An Introduction”, John Wiley & Sons, Inc. 1994.
2. P.C. Varghese, “Building Materials”, Prentice-Hall India, 2005.

V. REFERENCE BOOKS:

1. V. Raghavan, “Materials Science and Engineering”, Prentice Hall, 1990.
2. R.A. Higgins, “Properties of Engineering Materials”, Industrial Press, 1994.
3. Eds. J.M. Illston and P.L.J. Domone, “Construction Materials: Their nature and behaviour”, Spon Press, 3rd Edition, 2002

VI.WEB REFERENCES:

1. https://www.scribd.com/document/394619658/Material-Science-and-Engineering-V-Raghavan-pdf
2. https://files.isec.pt/DOCUMENTOS/SERVICOS/BIBLIO/INFORMA%C3%87%C3%95ES%20ADICIONAIS/Materials-for-engineers-5ed_Higgins.pdf

VII. E-TEXT BOOKS:

1. https://onlinecourses.nptel.ac.in/noc20_ce05/preview
2. http://kaizenha.com/wp-content/uploads/2016/04/Materials-Textbook-8th-Edition.pdf

DISASTER MANAGEMENT

<table>
<tr><td colspan="10">OE –I: VI Semester: AERO / MECH / CIVIL
OE – III: VIII Semester: CSE / CSE (AI & ML) / CSE (DS) / CSE (CS) / CSIT / IT / ECE / EEE</td></tr>
<tr><td>Course Code</td><td>Category</td><td colspan="3">Hours / Week</td><td>Credits</td><td colspan="3">Maximum Marks</td></tr>
<tr><td rowspan="2">ACEC31</td><td rowspan="2">Elective</td><td>L</td><td>T</td><td>P</td><td>C</td><td>CIA</td><td>SEE</td><td>Total</td></tr>
<tr><td>3</td><td>0</td><td>0</td><td>3</td><td>30</td><td>70</td><td>100</td></tr>
<tr><td>Contact Classes: 45</td><td>Total Tutorials: Nil</td><td colspan="4">Total Practical Classes: Nil</td><td colspan="3">Total Classes: 45</td></tr>
</table>

I. COURSE OVERVIEW:

The Disaster management provides a fundamental understanding of different aspects. It deals with the concepts and functions of disaster management to build competencies of professionals and development practitioners. It provides effective supporting environment by the governmental locating substantial resources for effective mitigation of disasters. It helps learners to apply the disaster mitigation strategies, preparedness for reducing damage intensity, loss of life and property.

II. COURSE OBJECTIVES:

The student will try to learn:

I. The concept of environmental hazards, disasters and various approaches dealing with the mitigation of disasters.
II. The knowledge on various types of environmental disasters and their impacts on human beings and nature.
III. The Different types of endogenous and exogenous hazards and their influence on human life and nature.
IV. The immediate response and damage assessment with information reporting and monitoring tools.

III.COURSE SYLLABUS:

MODULE-I: ENVIRONMENTAL HAZARDS AND DISASTERS (09)

Environmental hazards and disasters: meaning of environmental hazards, environmental disasters and environmental stress; concept of environmental hazards, environmental stress and environmental disasters, different approaches and relation with human ecology, landscape approach, ecosystem approach, perception approach, human ecology and its application in geographical researches.

MODULE-II: TYPES OF ENVIRONMENTAL HAZARDS AND DISASTERS (09)

Types of environmental hazards and disasters: Natural hazards and disasters, man induced hazards and disasters, natural hazards, planetary hazards/ disasters, extra planetary hazards/ disasters, planetary hazards, endogenous hazards, exogenous hazards.

MODULE-III: ENDOGENOUS HAZARDS (09)

Endogenous hazards, volcanic eruption, earthquakes, landslides, volcanic hazards/ disasters, causes and distribution of volcanoes, hazardous effects of volcanic eruptions, environmental impacts of volcanic eruptions.

Earthquake hazards/ disasters, causes of earthquakes, distribution of earthquakes, hazardous effects of, earthquakes, earthquake hazards in India, human adjustment, perception and mitigation of earthquake

MODULE-IV: EXOGENOUS HAZARDS (09)

Exogenous hazards/ disasters, infrequent events, cumulative atmospheric hazards/ disasters; Infrequent events: Cyclones , lightning , hailstorms; Cyclones: Tropical cyclones and local storms, destruction by tropical cyclones and local storms (causes, distribution human adjustment, perception and mitigation); Cumulative atmospheric hazards/ disasters: Floods, droughts, cold waves, heat waves floods; Causes of floods, flood hazards India, flood control measures (human adjustment, perception and mitigation); Droughts: Impacts of droughts, drought hazards in India, drought control measures, extra planetary hazards/ disasters, man induced hazards /disasters, physical hazards/ disasters, soil erosion, Soil erosion: Mechanics and forms of soil erosion, factors and causes of soil erosion, conservation measures of soil erosion; Chemical hazards/ disasters: Release of toxic chemicals, nuclear explosion, sedimentation processes; Sedimentation processes: Global sedimentation problems regional sedimentation problems, sedimentation and environmental problems, corrective measures of erosion and sedimentation, biological hazards/ disasters, population explosion.

MODULE-V: EMERGING APPROACHES IN DISASTER MANAGEMENT (09)

Emerging approaches in Disaster Management, Three Stages

1. Pre, disaster stage(preparedness)
2. Emergency Stage
3. Post Disaster stage, Rehabilitation.

IV.TEXT BOOKS:

1. Pardeep Sahni, "Disaster Mitigation: Experiences and Reflections", PHI Learning Pvt. Ltd., 1st Edition, 2001.
2. J.Glynn,Gary W.HeinKe, "Environmental Science and Engineering", Prentice Hall Publishers, 2nd Edition, 1996.

V. REFERENCE BOOKS:

1. B. C. Punmia, Ashok K Jain and Arun K Jain, "Mechanics of Materials", Laxmi Publications Pvt. Ltd., New Delhi, 12th Edition, 2007.
2. R. Subramanian, "Strength of Materials", Oxford University Press, 2nd Edition, 2010.
3. D. S. Prakash Rao, "Strength of Materials A Practical Approach Vol.1", Universities Press (India) Pvt. Ltd., India, 3rd Edition, 2007.
4. J. M. Gere, S.P. Timoshenko, "Mechanics of Materials, SI units edition", CL Engineering, USA, 5th Edition, 2000.

VI. Web References:

1. https://www.google.co.in/?gfe_rd=cr&ei=,iAwWLiDIazv8we8_5LADA#q=disater+mangement
2. http://ndma.gov.in/images/policyplan/dmplan/National%20Disaster%20Management%20Plan%20May%202016.pdf
3. http://www.eib.europa.eu/attachments/pipeline/20080021_eia_en.pdf
4. http://www.ndmindia.nic.in/

VII. E-Text Books:

1. http://cbse.nic.in/natural%20hazards%20&%20disaster%20management.pdf\
2. http://www.digitalbookindex.org/_search/search010emergencydisastera.asp
3. http://www.icbse.com/books/cbse,ebooks,download

INSTITUTE OF AERONAUTICAL ENGINEERING

(Autonomous)

Dundigal, Hyderabad - 500 043

UNDERTAKING BY STUDENT / PARENT

"To make the students attend the classes regularly from the first day of starting of classes and be aware of the College regulations, the following Undertaking Form is introduced which should be signed by both student and parent. The same should be submitted to the Dean of Academic".

I, Mr. / Ms. -- joining I Semester / III Semester for the academic year 20 - 20 / 20 - 20 in Institute of Aeronautical Engineering, Hyderabad, do hereby undertake and abide by the following terms, and I will bring the ACKNOWLEDGEMENT duly signed by me and my parent and submit it to the Dean of Academic.

1. I will attend all the classes as per the timetable from the starting day of the semester specified in the institute Academic Calendar. In case, I do not turn up even after two weeks of starting of classes, I shall be ineligible to continue for the current academic year.
2. I will be regular and punctual to all the classes (theory/laboratory/project) and secure attendance of not less than 75% in every course as stipulated by Institute. I am fully aware that an attendance of less than 65% in more than 60% of theory courses in a semester will make me lose one year.
3. I will compulsorily follow the dress code prescribed by the college.
4. I will conduct myself in a highly disciplined and decent manner both inside the classroom and on campus, failing which suitable action may be taken against me as per the rules and regulations of the institute.
5. I will concentrate on my studies without wasting time in the Campus/Hostel/Residence and attend all the tests to secure more than the minimum prescribed Class/Sessional Marks in each course. I will submit the assignments given in time to improve my performance.
6. I will not use Mobile Phone in the institute premises and also, I will not involve in any form of ragging inside or outside the campus. I am fully aware that using mobile phone to the institute premises is not permissible and involving in Ragging is an offence and punishable as per JNTUH/UGC rules and the law.
7. I declare that I shall not indulge in ragging, eve-teasing, smoking, consuming alcohol drug abuse or any other anti-social activity in the college premises, hostel, on educational tours, industrial visits or elsewhere.
8. I will pay tuition fees, examination fees and any other dues within the stipulated time as required by the Institution / authorities, failing which I will not be permitted to attend the classes.
9. I will not cause or involve in any sort of violence or disturbance both within and outside the college campus.
10. If I absent myself continuously for 3 days, my parents will have to meet the HOD concerned/ Principal.
11. I hereby acknowledge that I have received a copy of IARE – R20 Academic Rules and Regulations, Syllabus copy and hence, I shall abide by all the rules specified in it.

ACKNOWLEDGEMENT

I have carefully gone through the terms of the undertaking mentioned above and I understand that following these are for my/his/her own benefit and improvement. I also understand that if I/he/she fail to comply with these terms, shall be liable for suitable action as per Institute/JNTUH/AICTE/UGC rules and the law. I undertake that I/he/she will strictly follow the above terms.

Signature of Student with Date

Signature of Parent with Date
Name & Address with Phone Number