INSTITUTE OF AERONAUTICAL ENGINEERING

(Approved by AICTE, New Delhi, Affiliated to JNTUH, and Accredited by NBA)

Dundigal, Hyderabad – 500 043

OUTCOME BASED EDUCATION SYSTEM

B. Tech (R13 Regulations)

COMPUTER SCIENCE AND ENGINEERING

2015

Vision

The Vision of the Department is to produce competent graduates suitable for industries and organizations at global level including research and development with social responsibility.

Mission

To provide an open environment to foster professional and personal growth with a strong theoretical and practical background having an emphasis on hardware and software development making the graduates industry ready with social ethics. Further the Department is to provide training and to partner with global entities in education and research.

Contents

Part - I

PROGRAM EDUCATIONAL OBJECTIVES AND OUTCOMES

Genesis and Growth

The Department of Computer Science and Engineering was established in 2001 with a student intake of 40. The intake has been increased to 60 in 2002, 90 in 2004, 120 in 2006, 180 in 2012 and 240 in 2015. Besides, an additional 20% seats are under lateral entry scheme.

The UG program has been accredited by the National Board of Accreditation (NBA) since 2009.

The Computer Science and Engineering department at the institute is dedicated to providing educational opportunities in computer science related fields to specific undergraduate student body of talented boys and girls. The department emphasizes close interactions between students and the faculty dedicated to education and is actively engaged in events enriching the educational programs. The program emphasizes active learning with a strong laboratory component. The department nurtures intellectual, professional, and personal development of students with a view to transform them to competent professionals and responsible members of the society.

Computer Science and Engineering Department Advisory Council:

The Computer Science and Engineering Department Advisory Council (DAC) include a diverse group of experts from academe and industry, as well as alumni representation. The Advisory Board meets annually, or as needed, for a comprehensive review of the Computer Science and Engineering Department strategic planning and programs. The Advisory Council meets with the administration, faculty and students and prepares a report, which is presented to the principal. In each visit, the Department of Computer Science and Engineering responds to the DAC report indicating improvements and amendments to the program.

Program Educational Objectives, Outcomes and Assessment Criteria: The "Program Educational Objectives" were initially drafted by a committee of CSE faculty and were vetted and approved by a group of faculty from peer department, Information Technology and the CSE Department Advisory Council.

Goals

- To impart value-based education and motivate students to focus their efforts in the right direction.
- Design and equip the laboratories in the department to the emerging needs of the technology.
- Achieve 100% pass percentage in external examinations.
- Near 100% placement of all eligible students of the department.
- Establishing MoUs with reputed industries and universities for research, knowledge sharing and student placements.

- Ceaseless efforts to make the department a premier research and development center in the area of Big Data and Business Analytics.

1. PROGRAM EDUCATIONAL OBJECTIVES, OUTCOMES AND ASSESSMENT CRITERIA

The educational objectives of a module are statements of the broad intentions of the teaching team. They indicate what it is the teaching team intends to cover and the learning opportunities they intend to make available to the student. A learning outcome is a statement of what a learner (student) is expected to know, understand and/or be able to do at the end of a period of learning. It is advisable to express learning outcomes with the common prefix:

'On completion of (the period of learning e.g. module), the student **is expected to** be able to…'

Generally, learning outcomes do not specify curriculum, but more general areas of learning. It is not possible to prescribe precisely how specific a learning outcome statement should be. There is a balance to be struck between the degree of specificity in a learning outcome statement and that achieved by the assessment criteria (below).

If there are too many learning outcomes for a module, then either they are becoming assessment criteria or they are specifying too much curricular detail. The curriculum should be described in the range statement. Too few learning outcomes are unlikely to provide sufficient information on the course. As a guide, there should be between 3 and 4 learning outcomes for a course.

2. B.Tech - COMPUTER SCIENCE AND ENGINEERING PROGRAM EDUCATIONAL OBJECTIVES

A graduate of the Computer Science and Engineering Program should:

Program Educational Objective I
Students will establish themselves as effective professionals by solving real problems through the use of computer science knowledge and with attention to team work, effective communication, critical thinking and problem solving skills.

Program Educational Objective II
Students will develop professional skills that prepare them for immediate employment and for life-long learning in advanced areas of computer science and related fields.

Program Educational Objective III
Students will demonstrate their ability to adapt to a rapidly changing environment by having learned and applied new skills and new technologies.

Program Educational Objective IV
Students will be provided with an educational foundation that prepares them for excellence, leadership roles along diverse career paths with encouragement to professional ethics and active participation needed for a successful career.

These objectives are quite broad by intention, as Computer Science and Engineering graduates may seek further education or work in diverse areas. To make these objectives meaningful, they may be demonstrated by performance, actions, or achievements.

1. Students will establish themselves as effective professionals by solving real problems through the use of computer science knowledge and with attention to team work, effective communication, critical thinking and problem solving skills may be demonstrated by any of the following:

    - Acceptance by and satisfactory progress in a graduate degree program;
    - Significantly contributing to delivery of desired component, product, or process;
    - Formulating and solving moderately complex computer engineering problems, accounting for hardware/software/human interactions;
    - Skillfully using state-of-the-art tools for computer engineering processes;
    - Making practical recommendations that address computer engineering product and system level issues;
    - Producing clear written computer engineering documentation (papers, reports, and significant parts of proposals);
    - Communicating effectively in a group environment;
    - Being asked to make presentations or reports for internal colleagues or clients;
    - Publishing refereed paper in conference or journal, or producing an internally reviewed publication;
    - Making a significant contribution to a proposal;
    - Participating in the field through public speaking, activity in professional societies, technical associations, standards boards, etc;
    - Applying for a patent or making a useful invention;

2. Students will develop professional skills that prepare them for immediate employment and for life-long learning in advanced areas of computer science and related fields may be demonstrated by any of the following:

    - Successfully completing a course for B. Tech;
    - Successfully completing a tutorial at a conference;
    - Learning a new skill, tool, area, or system on your own;
    - Reading technical books, journals, conference papers, technical reports, or standards;
    - Attending a technical conference, symposium, or workshop;
    - Belonging to a professional society;

3. Students will demonstrate their ability to adapt to a rapidly changing environment by having learned and applied new skills and new technologies may be demonstrated by any of the following:

    - Appropriately using tools for collaboration, such as telecons, videocons, distributed meeting systems, etc;
    - Skillfully using tools for project and configuration management, e.g., resource planning systems, software source control systems, etc;
    - Making appropriate decisions on when to outsource, when to use off-the-shelf components, and when to develop components in-house;
    - Seeking assistance or elevating problems when necessary;
    - Properly handling a situation involving intellectual property rights;

4. Students will be provided with an educational foundation that prepares them for excellence, leadership roles along diverse career paths with encouragement to professional ethics and active participation needed for a successful career by the following any one:

    - Leading a project or design team;
    - Working successfully on ethnically, technically, or gender diverse teams;
    - Effectively resolving problems encountered in team work;
    - Estimating correctly the required resources (time, team, equipment, etc.) for computer engineering projects;
    - Promotion to managerial position;
    - Election or appointment to leadership position in a professional society;
    - Delegating effectively;
    - Participating in one of your organization's NSS programs;
    - Volunteering in a college, civic, or other charitable organization;
    - Participating in team sports or coaching;
    - Accounting for larger societal, legal, business, and technical context while making decisions on a project;
    - Properly handling a situation involving ethics;

3. B. Tech - PROGRAM OUTCOMES

A B. Tech program graduate will demonstrate:

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.

4. B. Tech - COMPUTER SCIENCE AND ENGINEERING PROGRAM SPECIFIC OUTCOMES

A graduate of the Computer Science and Engineering Program will demonstrate:

PSO1: **Professional Skills:** The ability to understand, analyze and develop computer programs in the areas related to algorithms, system software, multimedia, web design, big data analytics, and networking for efficient design of computer-based systems of varying complexity.

PSO2: **Problem-Solving Skills:** The ability to apply standard practices and strategies in software project development using open-ended programming environments to deliver a quality product for business success.

PSO3: **Successful Career and Entrepreneurship:** The ability to employ modern computer languages, environments, and platforms in creating innovative career paths to be an entrepreneur, and a zest for higher studies.

5. MAPPING OF PROGRAM EDUCATIONAL OBJECTIVES TO PROGRAM OUTCOMES

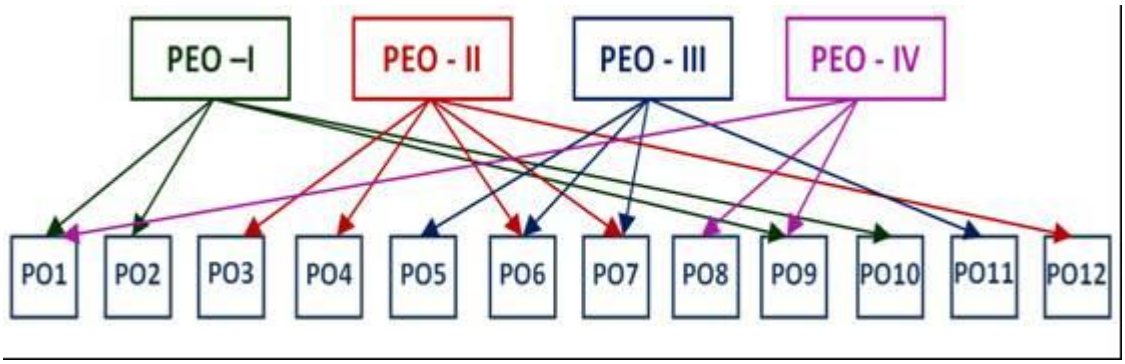
The details of the correlation between the program educational objectives and the program outcomes are tabulated as follows.

Program Educational Objective		***Program Outcomes***	
PEO I	*Students will establish themselves as effective professionals by solving real problems through the use of computer science knowledge and with attention to team work, effective communication, critical thinking and problem solving skills.*	PO1:	Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
		PO2:	Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

Program Educational Objective		Program Outcomes
		PO9: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings. PO10: 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.
PEO II	*Students will develop professional skills that prepare them for immediate employment and for life-long learning in advanced areas of computer science and related fields.*	PO3: 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: 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. PO6: 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: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development. PO12: 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.
PEO III	*Students will demonstrate their ability to adapt to a rapidly changing environment by having learned and applied new skills and new technologies.*	PO5: 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: 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: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

Program Educational Objective		*Program Outcomes*
		PO11: 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.
PEO IV	*Students will be provided with an educational foundation that prepares them for excellence, leadership roles along diverse career paths with encouragement to professional ethics and active participation needed for a successful career.*	PO1: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems. PO8: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice. PO9: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

Table 1: Correlation between the PEOS and the POs



The following figure shows the pictorial representation of the correlation between the program educational objectives and the program outcomes.

Figure 1: Correlation between the PEOs and the POs

A broad relation between the program educational objectives and program outcomes is given in the following table.

	PEO I: Effective Professionals by problem solving	**PEO II:** Lifelong learning in advanced areas of computer science and related fields	**PEO III:** Adapt to a rapidly changing environment by having learned and applied new skills and new technologies	**PEO IV:** Excellence and Leadership
PO1: Engineering knowledge	H	--	S	H

PO2: Problem analysis	H	--	S	--
PO3: Design/development of solutions	S	H	S	--
PO4: Conduct investigations of complex problems	S	H	--	--
PO5: Modern tool usage	--	S	H	--
PO6: The engineer and society	S	H	H	--
PO7: Environment and sustainability	S	H	H	--
PO8: Ethics	--	--	--	H
PO9: Individual and team work	H	--	--	H
PO10: Communication	H	--	--	S
PO11: Project management and finance	--	S	H	--
PO12: Life-long learning	--	H	--	S

Table 2: Broad **r**elationship between the PEOs and the POs
Key: H = Highly Related; S = Supportive

6. MAPPING OF PROGRAM EDUCATIONAL OBJECTIVES TO PROGRAM SPECIFIC OUTCOMES

The details of the correlation between the program educational objectives and the program specific outcomes are tabulated as follows.

Program Educational Objective		***Program Specific Outcomes***
PEO I	*Students will establish themselves as effective professionals by solving real problems through the use of computer science knowledge and with attention to team work, effective communication, critical thinking and problem solving skills.*	PSO2: The ability to apply standard practices and strategies in software project development using open-ended programming environments to deliver a quality product for business success.
PEO II	*Students will develop professional skills that prepare them for immediate employment and for life-long learning in advanced areas of computer science and related fields.*	PSO1: The ability to understand, analyze and develop computer programs in the areas related to algorithms, system software, multimedia, web design, big data analytics, and networking for efficient design of computer-based systems of varying complexity. PSO3: The ability to employ modern computer languages, environments, and platforms in creating innovative career paths to be an entrepreneur, and a zest for higher studies.
PEO III	*Students will demonstrate their ability to adapt to a rapidly changing environment by having learned and applied new skills and new technologies.*	PSO1: The ability to understand, analyze and develop computer programs in the areas related to algorithms, system software, multimedia, web design, big data analytics, and networking for efficient design of computer-based systems of varying complexity.

Program Educational Objective		*Program Specific Outcomes*
PEO IV	*Students will be provided with an educational foundation that prepares them for excellence, leadership roles along diverse career paths with encouragement to professional ethics and active participation needed for a successful career.*	PSO3: The ability to employ modern computer languages, environments, and platforms in creating innovative career paths, to be an entrepreneur and a zest for higher studies.

Table 3: Correlation between the PEOS and the PSOs

The following figure shows the pictorial representation of the correlation between the program educational objectives and the program specific outcomes.

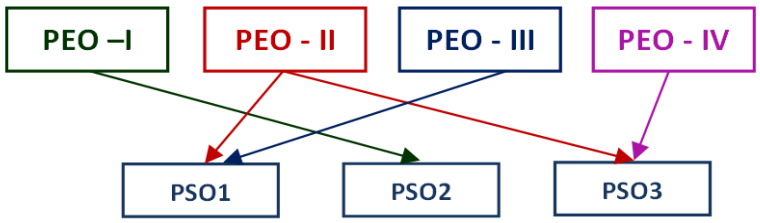


Figure 2: Correlation between the PEOs and the PSOs

A broad relation between the program educational objectives and program specific outcomes is given in the following table.

	PEO I: Effective Professionals by problem solving	**PEO II:** Professional skills that prepare them for immediate employment	**PEO III:** Adapt to a rapidly changing environment by having learned and applied new skills and new technologies	**PEO IV:** Active participation needed for a successful career
PSO1: Professional Skills	S	H	H	--
PSO2: Problem-Solving Skills	H	--	S	--
PSO3: Successful Career and Entrepreneurship	--	H	--	H

Table 4: Broad relationship between the PEOs and the PSOs
Key: H = Highly Related; S = Supportive

Note:

- The assessment process can be direct or indirect.
- The direct assessment will be through interim assessment by the faculty or by industry / technology experts.

- The indirect assessment on the other hand could be by students through course outcomes, lab evaluation, department associations, exit interviews, engineering services, GATE etc.
- Frequency of assessment can be once in a semester and justified by the programme coordinator.

7. SPECIFIC LEARNING OUTCOMES IN COMPUTER SCIENCE AND ENGINEERING

Our aim is to educate engineers who are ready to work actively in real-world, acquiring high ability in computer science and engineering discipline for the upcoming ubiquitous computing generation. Basic classes provided in the first two years give our students a concrete basis to be computer science engineers. Our students begin their professional study from the second year, and step up year by year to be highly-educated engineers. All our students obtain basic knowledge and basic skills in computer science and engineering discipline with the courses being offered in the undergraduate program and these courses nurture highly-educated engineers in information engineering, who have deep knowledge and problem-solving skills in the relevant fields. Our graduates are expected to have a wide range of abilities including professional knowledge and technical skills of software, hardware and web engineering, not only in basic mathematical informatics, to lead the world in IT industry.

Graduates of Computer Science and Engineering from accredited programmes must achieve the following learning outcomes, defined by the broad areas of learning.

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

- Knowledge and understanding of scientific principles and methodology necessary to strengthen their education in the computer science and engineering discipline, to enable appreciation of its scientific and engineering context, and to support their understanding of historical, current, and future developments and technologies in the field of computer science;
- Knowledge and understanding of mathematical principles necessary to underpin their education in the computer science and engineering discipline and to enable them to apply the same to solve real-world problems through automation;
- Ability to apply and integrate knowledge and understanding of other engineering disciplines to support their study in computer science and engineering discipline;
- Demonstrate understanding of basic concepts in computer science and engineering;
- Ability to integrate knowledge into computer systems in order to solve complex problems normally requiring a high level of human expertise;

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.

- Is based on the problem solving process that has been well documented in engineering texts. The elements of the process include:
    - Problem or opportunity identification

- Problem statement and system definition
- Problem formulation and abstraction
- Information and data collection

- Understanding of engineering principles and the ability to apply them to analyze key engineering processes;
- Ability to identify, classify and describe the performance of systems and components through the use of analytical methods and modeling techniques;
- Ability to apply quantitative methods and computer software relevant to the application environment, in order to solve real-world problems;
- Understanding of and ability to apply a systems approach to engineering problems;

As most engineers eventually learn, the problem solving process is never complete. Therefore, a final element here is feedback and improvement.

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.

- Design is the creation and development of an economically viable product, process or system to meet a defined need. It involves significant technical and intellectual challenges and needs to integrate all the engineering understanding, knowledge and skills in order to provide solutions of real-life problems. The elements of the design process include:
  - Modeling
  - Validation
  - Experimental design
  - Solution development or experimentation
  - Interpretation of results
  - Implementation and documentation

- Computer science and engineering graduates will therefore need the knowledge, understanding and skills to:
  - Investigate and define a problem and identify constraints including environmental and sustainability limitations, health and safety and risk assessment issues;
  - Understand the customer needs and the importance of considerations such as aesthetics, security etc;
  - Identify and manage cost drivers in the software development process;
  - Use creativity to establish innovative computer-based solutions;
  - Ensure fitness for purpose of all the aspects of the problem including production, operation, maintenance and disposal of the product;
  - Manage the design process and evaluate the outcomes;

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.

- The ability to pursue practical application of engineering skills, combining theory and experience, and use of other relevant knowledge and skills. This can include:

- Knowledge of the characteristics of the software-hardware interfaces, tools & technologies, processes, and products;
- Hands-on experience on the software design, development and testing tools;
- Understanding of contexts in which computer science engineering knowledge can be applied (for example, operations and management, technology development, etc.);
- Understanding of appropriate codes of practice and industry standards;
- Awareness of quality issues;
- Ability to work with technical uncertainty;

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.

- Encompasses a wide range of tools and skills needed by engineering graduates including computer software, simulation packages, diagnostic equipment and use of technical library resources and literature search tools;
- Ability to appreciate the ubiquitous support of modern tools in the software development of life cycle;
- Ability to accommodate with the ever emerging trend of tools and technologies in the software industry;

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.

- Here the focus is on "knowledge" and is interpreted to mean the student's obtaining in-depth knowledge of on contemporary issues. Three types of examples are given – socio economic, political and environmental excluding contemporary, technical engineering issues.
- Understanding use of technical literature and other information sources, awareness of nature of intellectual property and contractual issues;
- Awareness of quality issues;

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.

- Understanding of the requirement for computer science and engineering activities to promote sustainable development;
- Awareness of the framework of relevant legal requirements governing engineering activities, including personnel, health, safety, and risk (including environmental risk) issues;
- Assess the effects of the engineering products or solutions provided to solve real-world problems within the context of applicable environment;

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

- Understanding the need for a high level of professional and ethical conduct in engineering.

- Comprises four components: ability to make informed ethical choices, knowledge of professional codes of ethics, evaluation of the ethical dimensions of professional practice, and demonstration of the ethical behavior;
- Ability to confide in their beliefs;
- Ability to afford high degree of trust and integrity;

PO9: Individual and team work - Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

- Independence;
- Maturity – requiring only the achievement of goals to drive their performance;
- Self-direction (take a vaguely defined problem and systematically work to resolution);
- Teams are used during the classroom periods, in the hands-on labs, and in the design projects;
- Instruction on effective teamwork and project management is provided along with an appropriate textbook for reference;
- Teamwork is important not only for helping the students know their peers but also in completing assignments;
- Subjective evidence from senior students shows that the friendships and teamwork extends into the Junior years, and for some of those students, the friendships continues into the workplace after graduation;
- Ability to work with all levels of people in an organization;
- Ability to get along with others;
- Demonstrated ability to work well with a team;

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.

"Students should demonstrate the ability to communicate effectively in writing."

a. Clarity
b. Grammar/Punctuation
c. References

Verbal Communication: "Students should demonstrate the ability to communicate effectively orally."

a. Speaking Style
b. Subject Matter

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 as a leader in a team, to manage projects and in multidisciplinary environments.

- Knowledge of management techniques which may be used to achieve engineering objectives within that context;
- Knowledge and understanding of commercial and economic context of engineering processes;
- Understand the criteria in context of the product, application and users to deliver an effaceable project management process;

- Identify suitable management strategies and apply standard processes and procedures to achieve productive and conclusive effort;

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.

- Inspire the students to further explore in his/her program to recognize the need for life-long learning. Some aspects of life-long learning:
  - Concentrating on attainment of professional certifications;
  - Begin work to progress towards advanced degrees;
  - Being in step with the state-of-the-art technologies in the computer science and engineering field;
  - Personal continuing education efforts;
  - Continued personal development;

PSO1: Professional Skills - The ability to understand, analyze and develop computer programs in the areas related to algorithms, system software, multimedia, web design, big data analytics, and networking for efficient design of computer-based systems of varying complexity.

- Efficiency in designing and developing software programs, modifying existing computer software, testing software systems to meet requirements, documenting the work and consulting with other engineering professionals to assess the interface between the hardware and software;
- Ability to closely work with other information technology professionals including programmers, engineers and system analysts to analyze the specific requirements of users and to ensure that the software design is glued to the requirements analysis performed;
- Apart from developing new software programs, ability to work on the process of maintaining and enhancing the existing software programs for improved performance as and when needed;
- Ability to engage in appropriate discussions with the customers to comprehend the design requirements;

PSO2: Problem-Solving Skills - The ability to apply standard practices and strategies in software project development using open-ended programming environments to deliver a quality product for business success.

- Ability to work and coordinate tasks effectively in interdisciplinary teams;
- Analytical and critical thinking skills help a student to evaluate the problem and to make decisions. The student will be able to apply logical and methodical approach to come up with ideas for resolving problems;
- Ability to design, implement and evaluate alternative software solutions for modeling and simulating real-world activities.
- Ability to conduct in-depth research of the problem undertaken and provide conclusions with regard to the feasibilities in all aspects (for example, commercial aspects, financial aspects, consumer aspects etc);
- Capability to use a variety of scientific and mathematical techniques to assess the outcome of the software designs in various aspects such as quality, performance, fault tolerance, applicability, feasibility etc;

PSO3: Successful Career and Entrepreneurship - The ability to employ modern computer languages, environments, and platforms in creating innovative career paths to be an entrepreneur and a zest for higher studies.

- Create a plan for success that connects their college education to future career.
- The successful career will involve in work which is tightly integrated. Work that is an extension of your core being is work that is satisfying, fulfilling, meaningful, and enjoyable. Work meeting, a sure recipe for success.
- Graduates ready for immediate employment.
- Make a smooth transition into post graduate studies.

8. FACULTY OBJECTIVES

Each Faculty Member Should:

F1: Be able to teach a variety of Computer Science undergraduate courses.
F2: Actively participate in the undergraduate program.
F3: Strive to improve the quality of their teaching.
F4: Be aware of the recent developments in the fields that they normally teach so as to introduce undergraduates to recent research advances and practices.
F5: Should be active in the research fields where they formally teach and/or be involved in the field of computing education.
F6: Actively participate in the departmental, college, and university shared governance process.

9. COURSE STRUCTURE

A list of courses offered in Computer Science and Engineering curriculum (JNTUH-R13) and the achievable POs and PSOs, with effect from 01 June 2015 are tabulated as follows.

Code	Subject	Program Outcomes												Program Specific Outcomes		
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
I B. Tech																
A10001	English						X		X	X	X		X			X
A10002	Mathematics - I	X	X													
A10003	Mathematical Methods	X	X													
A10004	Engineering Physics	X	X					X								
A10005	Engineering Chemistry	X	X					X								
A10501	Computer Programming	X	X											X	X	
A10301	Engineering Drawing	X	X	X						X			X			
A10581	Computer Programming Lab		X	X									X	X	X	X
A10081	Engineering Physics/Engineering Chemistry Lab	X						X								
A10083	English Language Communication Skills Lab					X			X	X	X		X			X
A10082	IT Workshop/ Engineering Workshop	X				X							X	X		
II B. Tech I Semester																
A30008	Probability and Statistics	X	X													
A30504	Mathematical Foundations of Computer Science	X	X	X										X	X	
A30502	Data Structures	X	X	X										X	X	
A30401	Digital Logic Design	X	X	X										X	X	
A30404	Electronic Devices and Circuits	X	X													

Code	Subject	Program Outcomes												Program Specific Outcomes		
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
A30202	Basic Electrical Engineering	X	X													
A30282	Electrical and Electronics Lab		X							X						
A30582	Data Structures Lab		X	X	X	X							X	X	X	X
II B. Tech II Semester																
A40506	Computer Organization	X	X				X	X						X		
A40507	Database Management Systems	X	X	X	X		X		X					X	X	
A40503	Java Programming	X	X	X	X									X	X	
A40009	Environmental Studies							X	X				X			
A40509	Formal Languages and Automata Theory	X	X	X	X									X	X	
A40508	Design and Analysis of Algorithms	X	X	X	X		X							X	X	
A40585	Java Programming Lab		X	X	X	X							X	X	X	X
A40584	Database Management Systems Lab		X	X	X	X	X						X	X	X	X
III B. Tech I Semester																
A50511	Principles of Programming Languages	X	X											X		
A50018	Human Values and Professional Ethics (Open Elective)						X	X	X	X		X				X
A50017	Intellectual Property Rights (Open Elective)	X					X	X	X	X		X		X		
A50117	Disaster Management (Open Elective)	X					X	X	X	X		X				
A50518	Software Engineering	X	X				X	X		X		X		X	X	
A50514	Compiler Design	X	X	X										X		
A50510	Operating Systems	X	X				X	X						X		
A50515	Computer Networks	X	X	X			X	X					X	X	X	X

Code	Subject	Program Outcomes												Program Specific Outcomes		
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
A50589	Operating Systems Lab		X	X	X	X								X		X
A50587	Compiler Design Lab		X	X	X	X								X		X
III B. Tech II Semester																
A60521	Distributed Systems	X	X	X			X	X					X	X		X
A60522	Information Security	X	X	X			X	X	X				X	X	X	X
A60524	Object Oriented Analysis and Design	X	X	X			X							X	X	X
A60525	Software Testing Methodologies	X	X	X	X									X	X	X
A60010	Managerial Economics and Financial Analysis	X	X	X			X			X		X				X
A60512	Web Technologies	X	X	X									X	X	X	X
A60591	Case Tools and Web Technologies Lab		X	X	X	X							X	X	X	X
A60086	Advanced Communication Skills Lab					X			X	X	X		X			X
IV B. Tech I Semester																
A70511	Linux Programming	X	X	X										X	X	
A70530	Design Patterns	X	X	X										X	X	
A70520	Data Warehousing and Data Mining	X	X				X	X	X					X	X	
A70519	Cloud Computing	X	X				X	X					X	X	X	X
A70540	Software Project Management (Elective I)	X	X				X	X		X		X			X	X
A70532	Image Processing and Pattern Recognition (Elective I)	X	X				X	X						X		X
A70536	Mobile Computing (Elective I)	X	X				X	X					X	X	X	
A70529	Computer Graphics (Elective I)	X	X											X		
A70352	Operations Research (Elective I)	X	X							X		X				X

Code	Subject	Program Outcomes												Program Specific Outcomes		
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
A70534	Machine Learning (Elective II)	X	X				X	X	X					X	X	
A70539	Soft Computing (Elective II)	X	X				X	X						X	X	
A70533	Information Retrieval Systems (Elective II)	X	X											X	X	
A70526	Artificial Intelligence (Elective II)	X	X	X			X	X					X	X	X	
A70628	Computer Forensics (Elective II)	X	X				X	X	X							
A70596	Linux Programming Lab		X	X	X	X							X	X	X	X
A70595	Data Warehousing and Mining Lab		X	X	X	X							X	X	X	X
IV B. Tech II Semester																
A80014	Management Science	X	X	X			X	X		X		X				X
A80551	Web Services (Elective III)	X	X	X									X	X	X	X
A80538	Semantic Web and Social Networks (Elective III)	X					X	X	X				X	X	X	X
A80537	Scripting Languages (Elective III)	X		X										X		
A80547	Multimedia and Rich Internet Applications (Elective III)	X	X	X									X	X		X
A80542	Ad hoc and Sensor Networks (Elective IV)	X	X										X	X	X	X
A80550	Storage Area Networks (Elective IV)	X	X										X	X	X	X
A80543	Database Security (Elective IV)	X	X				X	X	X				X	X	X	X
A80439	Embedded Systems (Elective IV)	X	X	X	X								X	X	X	X
A80087	Industry Oriented Mini Project	X	X	X	X	X				X	X	X		X	X	
A80089	Seminar	X									X			X		
A80088	Project Work	X	X	X	X	X				X	X	X	X	X	X	
A80090	Comprehensive Viva	X									X			X		

Table 5: Course Curriculum (JNTUH - R13) and the achievable POs and PSOs

10. OUTCOME DELIVERY AND ASSESSMENT

The categorization of the Computer Science and Engineering courses offered in JNTUH-R13 curriculum that support the achievement of each of the program outcomes and the program specific outcomes is as follows.

Code	Subject	Code	Subject
PO1: Engineering knowledge - Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.			
A10002	Mathematics - I	A60010	Managerial Economics and Financial Analysis
A10003	Mathematical Methods	A60512	Web Technologies
A10004	Engineering Physics	A70511	Linux Programming
A10005	Engineering Chemistry	A70530	Design Patterns
A10501	Computer Programming	A70520	Data Warehousing and Data Mining
A10301	Engineering Drawing	A70519	Cloud Computing
A10081	Engineering Physics/Engineering Chemistry Lab	A70540	Software Project Management (Elective I)
A10082	IT Workshop/ Engineering Workshop	A70532	Image Processing and Pattern Recognition (Elective I)
A30008	Probability and Statistics	A70536	Mobile Computing (Elective I)
A30504	Mathematical Foundations of Computer Science	A70529	Computer Graphics (Elective I)
A30502	Data Structures	A70352	Operations Research (Elective I)
A30401	Digital Logic Design	A70534	Machine Learning (Elective II)
A30404	Electronic Devices and Circuits	A70539	Soft Computing (Elective II)
A30202	Basic Electrical Engineering	A70533	Information Retrieval Systems (Elective II)
A40506	Computer Organization	A70526	Artificial Intelligence (Elective II)
A40507	Database Management Systems	A70628	Computer Forensics (Elective II)
A40503	Java Programming	A80014	Management Science
A40509	Formal Languages and Automata Theory	A80551	Web Services (Elective III)
A40508	Design and Analysis of Algorithms	A80538	Semantic Web and Social Networks (Elective III)
A50511	Principles of Programming Languages	A80537	Scripting Languages (Elective III)
A50017	Intellectual Property Rights (Open Elective)	A80547	Multimedia and Rich Internet Applications (Elective III)
A50117	Disaster Management (Open Elective)	A80542	Ad hoc and Sensor Networks (Elective IV)
A50518	Software Engineering	A80550	Storage Area Networks (Elective IV)
A50514	Compiler Design	A80543	Database Security (Elective IV)
A50510	Operating Systems	A80439	Embedded Systems (Elective IV)
A50515	Computer Networks	A80087	Industry Oriented Mini Project
A60521	Distributed Systems	A80089	Seminar
A60522	Information Security	A80088	Project Work
A60524	Object Oriented Analysis and Design	A80090	Comprehensive Viva
A60525	Software Testing Methodologies		
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.			
A10002	Mathematics - I	A60524	Object Oriented Analysis and Design
A10003	Mathematical Methods	A60525	Software Testing Methodologies
A10004	Engineering Physics	A60010	Managerial Economics and Financial Analysis

Code	Subject	Code	Subject
A10005	Engineering Chemistry	A60512	Web Technologies
A10501	Computer Programming	A60591	Case Tools and Web Technologies Lab
A10301	Engineering Drawing	A70511	Linux Programming
A10581	Computer Programming Lab	A70530	Design Patterns
A30008	Probability and Statistics	A70520	Data Warehousing and Data Mining
A30504	Mathematical Foundations of Computer Science	A70519	Cloud Computing
A30502	Data Structures	A70540	Software Project Management (Elective I)
A30401	Digital Logic Design	A70532	Image Processing and Pattern Recognition (Elective I)
A30404	Electronic Devices and Circuits	A70536	Mobile Computing (Elective I)
A30202	Basic Electrical Engineering	A70529	Computer Graphics (Elective I)
A30282	Electrical and Electronics Lab	A70352	Operations Research (Elective I)
A30582	Data Structures Lab	A70534	Machine Learning (Elective II)
A40506	Computer Organization	A70539	Soft Computing (Elective II)
A40507	Database Management Systems	A70533	Information Retrieval Systems (Elective II)
A40503	Java Programming	A70526	Artificial Intelligence (Elective II)
A40509	Formal Languages and Automata Theory	A70628	Computer Forensics (Elective II)
A40508	Design and Analysis of Algorithms	A70596	Linux Programming Lab
A40585	Java Programming Lab	A70595	Data Warehousing and Mining Lab
A40584	Database Management Systems Lab	A80014	Management Science
A50511	Principles of Programming Languages	A80551	Web Services (Elective III)
A50518	Software Engineering	A80547	Multimedia and Rich Internet Applications (Elective III)
A50514	Compiler Design	A80542	Ad hoc and Sensor Networks (Elective IV)
A50510	Operating Systems	A80550	Storage Area Networks (Elective IV)
A50515	Computer Networks	A80543	Database Security (Elective IV)
A50589	Operating Systems Lab	A80439	Embedded Systems (Elective IV)
A50587	Compiler Design Lab	A80087	Industry Oriented Mini Project
A60521	Distributed Systems	A80088	Project Work
A60522	Information Security		
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.			
A10301	Engineering Drawing	A60524	Object Oriented Analysis and Design
A10581	Computer Programming Lab	A60525	Software Testing Methodologies
A30504	Mathematical Foundations of Computer Science	A60010	Managerial Economics and Financial Analysis
A30502	Data Structures	A60512	Web Technologies
A30401	Digital Logic Design	A60591	Case Tools and Web Technologies Lab
A30582	Data Structures Lab	A70511	Linux Programming
A40507	Database Management Systems	A70530	Design Patterns
A40503	Java Programming	A70526	Artificial Intelligence (Elective II)
A40509	Formal Languages and Automata Theory	A70596	Linux Programming Lab
A40508	Design and Analysis of Algorithms	A70595	Data Warehousing and Mining Lab
A40585	Java Programming Lab	A80014	Management Science

Code	Subject	Code	Subject
A40584	Database Management Systems Lab	A80551	Web Services (Elective III)
A50514	Compiler Design	A80537	Scripting Languages (Elective III)
A50515	Computer Networks	A80547	Multimedia and Rich Internet Applications (Elective III)
A50589	Operating Systems Lab	A80439	Embedded Systems (Elective IV)
A50587	Compiler Design Lab	A80087	Industry Oriented Mini Project
A60521	Distributed Systems	A80088	Project Work
A60522	Information Security		
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.			
A30582	Data Structures Lab	A50587	Compiler Design Lab
A40507	Database Management Systems	A60525	Software Testing Methodologies
A40503	Java Programming	A60591	Case Tools and Web Technologies Lab
A40509	Formal Languages and Automata Theory	A70596	Linux Programming Lab
A40508	Design and Analysis of Algorithms	A70595	Data Warehousing and Mining Lab
A40585	Java Programming Lab	A80439	Embedded Systems (Elective IV)
A40584	Database Management Systems Lab	A80087	Industry Oriented Mini Project
A50589	Operating Systems Lab	A80088	Project Work
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.			
A10083	English Language Communication Skills Lab	A60591	Case Tools and Web Technologies Lab
A10082	IT Workshop/ Engineering Workshop	A60086	Advanced Communication Skills Lab
A30582	Data Structures Lab	A70596	Linux Programming Lab
A40585	Java Programming Lab	A70595	Data Warehousing and Mining Lab
A40584	Database Management Systems Lab	A80087	Industry Oriented Mini Project
A50589	Operating Systems Lab	A80088	Project Work
A50587	Compiler Design Lab		
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.			
A10001	English	A60010	Managerial Economics and Financial Analysis
A40506	Computer Organization	A70520	Data Warehousing and Data Mining
A40507	Database Management Systems	A70519	Cloud Computing
A40508	Design and Analysis of Algorithms	A70540	Software Project Management (Elective I)
A40584	Database Management Systems Lab	A70532	Image Processing and Pattern Recognition (Elective I)
A50018	Human Values and Professional Ethics (Open Elective)	A70536	Mobile Computing (Elective I)
A50017	Intellectual Property Rights (Open Elective)	A70534	Machine Learning (Elective II)
A50117	Disaster Management (Open Elective)	A70539	Soft Computing (Elective II)
A50518	Software Engineering	A70526	Artificial Intelligence (Elective II)
A50510	Operating Systems	A70628	Computer Forensics (Elective II)
A50515	Computer Networks	A80014	Management Science
A60521	Distributed Systems	A80538	Semantic Web and Social Networks (Elective III)

Code	Subject	Code	Subject
A60522	Information Security	A80543	Database Security (Elective IV)
A60524	Object Oriented Analysis and Design		
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.			
A10004	Engineering Physics	A70520	Data Warehousing and Data Mining
A10005	Engineering Chemistry	A70519	Cloud Computing
A10081	Engineering Physics/Engineering Chemistry Lab	A70540	Software Project Management (Elective I)
A40506	Computer Organization	A70532	Image Processing and Pattern Recognition (Elective I)
A40009	Environmental Studies	A70536	Mobile Computing (Elective I)
A50018	Human Values and Professional Ethics (Open Elective)	A70534	Machine Learning (Elective II)
A50017	Intellectual Property Rights (Open Elective)	A70539	Soft Computing (Elective II)
A50117	Disaster Management (Open Elective)	A70526	Artificial Intelligence (Elective II)
A50518	Software Engineering	A70628	Computer Forensics (Elective II)
A50510	Operating Systems	A80014	Management Science
A50515	Computer Networks	A80538	Semantic Web and Social Networks (Elective III)
A60521	Distributed Systems	A80543	Database Security (Elective IV)
A60522	Information Security		
PO8: Ethics - Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.			
A10001	English	A60522	Information Security
A10083	English Language Communication Skills Lab	A60086	Advanced Communication Skills Lab
A40507	Database Management Systems	A70520	Data Warehousing and Data Mining
A40009	Environmental Studies	A70534	Machine Learning (Elective II)
A50018	Human Values and Professional Ethics (Open Elective)	A70628	Computer Forensics (Elective II)
A50017	Intellectual Property Rights (Open Elective)	A80538	Semantic Web and Social Networks (Elective III)
A50117	Disaster Management (Open Elective)	A80543	Database Security (Elective IV)
PO9: Individual and team work - Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.			
A10001	English	A60010	Managerial Economics and Financial Analysis
A10301	Engineering Drawing	A60086	Advanced Communication Skills Lab
A10083	English Language Communication Skills Lab	A70540	Software Project Management (Elective I)
A30282	Electrical and Electronics Lab	A70352	Operations Research (Elective I)
A50018	Human Values and Professional Ethics (Open Elective)	A80014	Management Science
A50017	Intellectual Property Rights (Open Elective)	A80087	Industry Oriented Mini Project
A50117	Disaster Management (Open Elective)	A80088	Project Work
A50518	Software Engineering		
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.			
A10001	English	A80089	Seminar
A10083	English Language Communication Skills Lab	A80088	Project Work
A60086	Advanced Communication Skills Lab	A80090	Comprehensive Viva
A80087	Industry Oriented Mini Project		

Code	Subject	Code	Subject
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.			
A50018	Human Values and Professional Ethics (Open Elective)	A70540	Software Project Management (Elective I)
A50017	Intellectual Property Rights (Open Elective)	A70352	Operations Research (Elective I)
A50117	Disaster Management (Open Elective)	A80014	Management Science
A50518	Software Engineering	A80087	Industry Oriented Mini Project
A60010	Managerial Economics and Financial Analysis	A80088	Project Work
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.			
A10001	English	A60086	Advanced Communication Skills Lab
A10301	Engineering Drawing	A70519	Cloud Computing
A10581	Computer Programming Lab	A70536	Mobile Computing (Elective I)
A10083	English Language Communication Skills Lab	A70526	Artificial Intelligence (Elective II)
A10082	IT Workshop/ Engineering Workshop	A70596	Linux Programming Lab
A30582	Data Structures Lab	A70595	Data Warehousing and Mining Lab
A40009	Environmental Studies	A80551	Web Services (Elective III)
A40585	Java Programming Lab	A80538	Semantic Web and Social Networks (Elective III)
A40584	Database Management Systems Lab	A80547	Multimedia and Rich Internet Applications (Elective III)
A50515	Computer Networks	A80542	Ad hoc and Sensor Networks (Elective IV)
A60521	Distributed Systems	A80550	Storage Area Networks (Elective IV)
A60522	Information Security	A80543	Database Security (Elective IV)
A60512	Web Technologies	A80439	Embedded Systems (Elective IV)
A60591	Case Tools and Web Technologies Lab	A80088	Project Work
PSO1: Professional Skills - The ability to understand, analyze and develop computer programs in the areas related to algorithms, system software, multimedia, web design, big data analytics, and networking for efficient design of computer-based systems of varying complexity.			
A10501	Computer Programming	A60591	Case Tools and Web Technologies Lab
A10581	Computer Programming Lab	A70511	Linux Programming
A10082	IT Workshop/ Engineering Workshop	A70530	Design Patterns
A30504	Mathematical Foundations of Computer Science	A70520	Data Warehousing and Data Mining
A30502	Data Structures	A70519	Cloud Computing
A30401	Digital Logic Design	A70540	Software Project Management (Elective I)
A30582	Data Structures Lab	A70532	Image Processing and Pattern Recognition (Elective I)
A40506	Computer Organization	A70536	Mobile Computing (Elective I)
A40507	Database Management Systems	A70529	Computer Graphics (Elective I)
A40503	Java Programming	A70534	Machine Learning (Elective II)
A40509	Formal Languages and Automata Theory	A70539	Soft Computing (Elective II)
A40508	Design and Analysis of Algorithms	A70533	Information Retrieval Systems (Elective II)
A40585	Java Programming Lab	A70526	Artificial Intelligence (Elective II)
A40584	Database Management Systems Lab	A70596	Linux Programming Lab
A50511	Principles of Programming Languages	A70595	Data Warehousing and Mining Lab
A50017	Intellectual Property Rights (Open Elective)	A80551	Web Services (Elective III)

Code	Subject	Code	Subject
A50518	Software Engineering	A80538	Semantic Web and Social Networks (Elective III)
A50514	Compiler Design	A80537	Scripting Languages (Elective III)
A50510	Operating Systems	A80547	Multimedia and Rich Internet Applications (Elective III)
A50515	Computer Networks	A80542	Ad hoc and Sensor Networks (Elective IV)
A50589	Operating Systems Lab	A80550	Storage Area Networks (Elective IV)
A50587	Compiler Design Lab	A80543	Database Security (Elective IV)
A60521	Distributed Systems	A80439	Embedded Systems (Elective IV)
A60522	Information Security	A80087	Industry Oriented Mini Project
A60524	Object Oriented Analysis and Design	A80088	Project Work
A60525	Software Testing Methodologies	A80089	Seminar
A60512	Web Technologies	A80090	Comprehensive Viva
PSO2: Problem-Solving Skills - The ability to apply standard practices and strategies in software project development using open-ended programming environments to deliver a quality product for business success.			
A10501	Computer Programming	A70530	Design Patterns
A10581	Computer Programming Lab	A70520	Data Warehousing and Data Mining
A30504	Mathematical Foundations of Computer Science	A70519	Cloud Computing
A30502	Data Structures	A70540	Software Project Management (Elective I)
A30401	Digital Logic Design	A70536	Mobile Computing (Elective I)
A30582	Data Structures Lab	A70534	Machine Learning (Elective II)
A40507	Database Management Systems	A70539	Soft Computing (Elective II)
A40503	Java Programming	A70533	Information Retrieval Systems (Elective II)
A40509	Formal Languages and Automata Theory	A70526	Artificial Intelligence (Elective II)
A40508	Design and Analysis of Algorithms	A70596	Linux Programming Lab
A40585	Java Programming Lab	A70595	Data Warehousing and Mining Lab
A40584	Database Management Systems Lab	A80551	Web Services (Elective III)
A50518	Software Engineering	A80538	Semantic Web and Social Networks (Elective III)
A50515	Computer Networks	A80542	Ad hoc and Sensor Networks (Elective IV)
A60522	Information Security	A80550	Storage Area Networks (Elective IV)
A60524	Object Oriented Analysis and Design	A80543	Database Security (Elective IV)
A60525	Software Testing Methodologies	A80439	Embedded Systems (Elective IV)
A60512	Web Technologies	A80087	Industry Oriented Mini Project
A60591	Case Tools and Web Technologies Lab	A80088	Project Work
A70511	Linux Programming		
PSO3: Successful Career and Entrepreneurship - The ability to employ modern computer languages, environments, and platforms in creating innovative career paths, to be an entrepreneur, and a zest for higher studies.			
A10001	English	A60591	Case Tools and Web Technologies Lab
A10581	Computer Programming Lab	A60086	Advanced Communication Skills Lab
A10083	English Language Communication Skills Lab	A70519	Cloud Computing
A30582	Data Structures Lab	A70540	Software Project Management (Elective I)
A40585	Java Programming Lab	A70532	Image Processing and Pattern Recognition (Elective I)
A40584	Database Management Systems Lab	A70352	Operations Research (Elective I)
A50018	Human Values and Professional Ethics (Open Elective)	A70596	Linux Programming Lab

Code	Subject	Code	Subject
A50515	Computer Networks	A70595	Data Warehousing and Mining Lab
A50589	Operating Systems Lab	A80014	Management Science
A50587	Compiler Design Lab	A80551	Web Services (Elective III)
A60521	Distributed Systems	A80538	Semantic Web and Social Networks (Elective III)
A60522	Information Security	A80547	Multimedia and Rich Internet Applications (Elective III)
A60524	Object Oriented Analysis and Design	A80542	Ad hoc and Sensor Networks (Elective IV)
A60525	Software Testing Methodologies	A80550	Storage Area Networks (Elective IV)
A60010	Managerial Economics and Financial Analysis	A80543	Database Security (Elective IV)
A60512	Web Technologies	A80439	Embedded Systems (Elective IV)

Table 6: Categorization of the CSE courses (JNTUH R13) based on their support in the achievement POs and PSOs

11. METHODS OF MEASURING LEARNING OUTCOMES AND VALUE ADDED

Methodologies that are used to measure student learning each have their own limitations and biases, and no method can be counted on to be completely error free. That is why best practice in educational research dictates triangulating the data. If several different sources of data are used, it increases the probability that the findings present an accurate picture. We employ the following formal assessment procedures:

1) End-of-semester course evaluations
2) Departmental mid-semester course evaluations
3) Departmental course objective surveys
4) Course portfolio evaluations
5) Exit Interviews
6) Alumni feedback
7) Employer surveys
8) Department academic council meetings
9) Faculty meetings
10) Project work
11) Job Placements
12) Professional societies

Each is described in more detail below:

1) ***University end-of-semester course evaluations:*** J N T University conducts end-of-semester examination for all courses. Summary results for each course are distributed to the appropriate instructor and the HOD, summarizing the course-specific results and comparing them to the average across the university. Students are encouraged to write specific comments about the positive and negative aspects of the course. The statistical summary and student comments presented are submitted to the principal and department academic council for review.

2) ***Departmental mid-semester course evaluations:*** The Computer Science and Engineering department conducts mid-semester reviews for all courses. All departmental students are encouraged to fill out a brief survey on the state of the courses they are currently taking, and space is provided for a written comment. Faculty are strongly encouraged to review

these evaluations, and draft a brief response on how they will react to correct any deficiencies noted by the students. The results are reviewed by the departmental faculty (all faculty have permission to read results for all courses).

3) ***Departmental course objective surveys:*** The Computer Science and Engineering department conducts end-of-semester course objective surveys for all of our courses. All departmental students are encouraged to fill out a brief survey on the state of the courses they are currently taking, and space is provided for a written comment. Faculty are strongly encouraged to review these evaluations, and draft a brief response on how they will react to correct any deficiencies noted by the students. The results are reviewed by the departmental faculty (all faculty have permission to read results for all courses). The results of how courses satisfy their objectives are discussed at a faculty meeting.

4) ***Course portfolio evaluations:*** We collect course portfolios of each course offered in the given semester from the instructor. They remain on file for our entire faculty to study. These portfolios help the course coordinator monitor how the course is being taught, and help new faculty understand how more experienced colleagues teach the given course. With respect to the assessment, each portfolio contains two surveys to be filled out by the instructor of the course. The beginning-of-semester survey encourages faculty members to think about what they can do to improve the teaching and administration of their course, compared with the last time they taught it. The end-of-semester survey encourages faculty to record what did and did not work well during this course offering and what changes should be made for the future.

5) ***Exit Interviews:*** Inputs from final year students are solicited annually through Computer Science and Engineering Exit Survey. The results are disseminated to the faculty and department advisory council for analysis and discussion. The questionnaire is designed to survey program outcomes, solicit about program experiences, career choices as well as suggestions and comments. This instrument seeks to assess how students view the department's program in retrospect.

6) ***Alumni feedback:*** The alumni survey is a written questionnaire which alumni are asked to complete. We use this survey seeking input on the Program Objectives and Learning Outcomes based on their experience after graduation and after they have spent time in the working world. Alumni are an excellent resource with perspective on the value and advantages of their education. They are also resource for current students for potential networking and employment. The data will be analyzed and used in continuous improvement.

7) ***Employer surveys:*** The employer survey is a written questionnaire which employers of the program's graduates are asked to complete. We review the effectiveness of our curriculum and how well the student is prepared in the department of Computer Science and Engineering, VCE. To do this, we survey Employers and Advisors of alumni who graduated four years ago. We ask about several categories of preparation, and for each category, how well does they think he or she was prepared, and how important you think preparation in that area is to him or her in the current position. This survey will greatly assist us in determining the college overall level of achievement of our Program Educational Objectives.

8) ***Department academic council meetings:*** The Computer Science and Engineering Department Advisory Council (CSEDAC) include a diverse group of experts from academe and industry, as well as alumni representation. The Advisory Board meets annually, or as needed, for a comprehensive review of the Computer Science and Engineering department strategic planning and programs. The Advisory Council meets with administration, faculty and students and prepares a report, which is presented to

principal. In each visit, the Department of Computer Science and Engineering responds to the report indicating improvements and amendments to the program.

9) ***Faculty meetings:*** The state of undergraduate program is always on the agenda at the monthly meeting of the faculty. The faculty devotes a substantial amount of time to formal and informal discussions assessing the state of program and searching for improvements.

10) ***Project work:*** The final project reports, must demonstrate that students produced solutions to research/industry problems involving contemporary issues. There is no scale for this tool as the reports provide qualitative data.

11) ***Job Placements:*** Data from the Placement and Training Centre on graduates' job placement reflects how successful our graduates are in securing a job in a related field.

12) ***Professional societies:*** The role of professional societies in introducing our students to technical, entrepreneurial and societal aspects of the field and in providing outstanding opportunities for lifelong learning makes them important constituents. Department of Computer Science and Engineering supports CSI student chapters and encourages student participation as a means for enhancing the profession, networking and leadership skills.

Part - II

WRITING AND ASSESSING COURSE LEVEL STUDENT LEARNING OUTCOMES

Although the term "Expected Learning Outcome" may be new, the process of identifying the key concepts or skills that students are expected to learn during specific courses is not. Many people are more familiar with the terms "course objective" or "course competency". Expected learning outcomes are really very similar to both of these concepts. So if course objectives or competencies are available, the process of having expected learning outcomes for class is closer.

This will provide information on exactly what expected learning outcomes are and what methods can be used to assess them. This is designed to assist faculty with the process of developing expected learning outcomes and methods for assessing those outcomes in their courses. This provides basic information related to (1) course purpose; (2) expected learning outcomes; (3) methods for assessing expected learning outcomes; (4) criteria for grade determination; and (5) a course outline.

Expected Learning Outcomes: *After reading and completing this, individuals will be able to:*

- Prepare a description of the course as well as a written statement regarding the course's purpose;
- Construct/develop expected learning outcomes for the course;
- Create an assessment plan that outlines the specific methods that will be used to assess the expected student learning outcomes for a course;
- Describe how grades will be determined in a process that is separate and distinct from assessing the expected learning outcomes;
- Identify the common components of a course outline ;
- Revise their course syllabi to incorporate a course purpose, expected learning outcomes, methods to assess those outcomes, the criteria for grade determination, and a course outline;
- This process uses some terminology related to the expected learning outcomes and assessment. A brief glossary of terms has been provided below for reference purposes;

Assessment of expected learning outcomes: The process of investigating

a) What students are learning?
b) How well they are learning it in relation to the stated expected learning outcomes for the course?

Assessment plan: The proposed methods and timeline for assessment-related activities in a given course (e.g., when are you going to check what/how well the students are learning and how are you going to do that?).

Classroom Assessment Technique (CAT): Angelo and Cross (1993) developed a variety of techniques/activities than can be used to assess students' learning. These CATs are often done anonymously and are not graded. These activities check on the class' learning while students are still engaged in the learning process. An example of a CAT is a non-graded quiz given a few weeks before the first exam.

Course description: A formal description of the material to be covered in the course.

Course purpose: The course purpose describes the intent of the course and how it contributes to the programme. The course purpose goes beyond the course description.

Expected learning outcome: A formal statement of what students are expected to learn in a course (synonyms for "expected learning outcome" include learning outcome, learning outcome statement, and student learning outcome).

Evaluation: Making a judgment about the quality of student's learning/work and assigning marks based on that judgment. Evaluation activities (such as exams, papers, etc.) are often seen as formal ways to assess the expected learning outcomes for a course.

Methods for assessing student learning outcomes: This term refers to any technique or activity that is used to identify what students are learning or how well they are learning. Formal methods for evaluating student learning outcomes include Continuous Assessment Tests, Mid Semester Test, Tutorials, End Semester Examination etc. The assessment methods are used to identify how the well students have acquired the learning outcomes for the course.

1. COURSE PURPOSE

One of the first steps in identifying the expected learning outcomes for a course is identifying the *purpose* of teaching in the course. By clarifying the purpose of the course, faculty can help discover the main topics or themes related to students' learning. These themes help to outline the expected learning outcomes for the course.

The course purpose involves the following:

a) What role does this course play within the programme?
b) How is the course unique or different from other courses?
c) Why should/do students take this course? What essential knowledge or skills should they gain from this experience?
d) What knowledge or skills from this course will students need to have mastered to perform well in future classes or jobs?
e) Why is this course important for students to take?

The "Course Description" provides general information regarding the topics and content addressed in the course, the "Course Purpose" goes beyond that to describe how this course fits in to the students' educational experience in the programme.

2. EXPECTED LEARNING OUTCOMES

Expected Learning Outcome (definition)

An expected learning outcome is a formal statement of what students are expected to learn in a course. Expected learning outcome statements refer to specific knowledge, practical skills, areas of professional development, attitudes, higher-order thinking skills, etc. that faculty members expect students to develop, learn, or master during a course (Suskie, 2004). Expected learning outcomes are also often referred to as "learning outcomes", "student learning outcomes", or "learning outcome statements".

Simply stated, expected learning outcome statements describe:

1. What faculty members want students to *know* at the end of the course?

2. What faculty members want students *to be able to do* at the end of the course?

Learning outcomes have three major characteristics

1. They specify an action by the students/learners that is ***observable;***
2. They specify an action by the students/learners that is ***measurable;***
3. They specify an action that is done by the ***students/learners*** (rather than the faculty members);

Effectively developed expected learning outcome statements should possess all three of these characteristics. When this is done, the expected learning outcomes for a course are designed so that they can be assessed (Suskie, 2004).

3. WRITING EFFECTIVE LEARNING OUTCOME STATEMENTS

When stating expected learning outcomes, it is important to use verbs that describe exactly what the learner(s) will be able to *do* upon completion of the course.

Examples of good action words to include in expected learning outcome statements:

Compile, identify, create, plan, revise, analyze, design, select, utilize, apply, demonstrate, prepare, use, compute, discuss, explain, predict, assess, compare, rate, critique, outline, or evaluate

There are some verbs that are unclear in the context of an expected learning outcome statement (*e.g., know, be aware of, appreciate, learn, understand, comprehend*). These words are often vague, have multiple interpretations, or are simply difficult to observe or measure (American Association of Law Libraries, 2005). As such, it is best to avoid using these terms when creating expected learning outcome statements.

For example, please look at the following learning outcomes statements:

- *The students will understand basic Data Mining techniques.*
- *The students will appreciate knowledge discovery from Data Mining techniques.*

Both of these learning outcomes are stated in a manner that will make them difficult to assess. Consider the following:

- How do you observe someone "understanding" a theory or "appreciating" Data Mining techniques?
- How easy will it be to measure "understanding" or "appreciation"?

These expected learning outcomes are more effectively stated the following way:

- *The students will be able to identify and describe what techniques are used to extract knowledge from Database Repositories.*
- *The students will be able to identify the characteristics of Classification techniques from other Data Mining techniques.*

Incorporating Critical Thinking Skills into Expected Learning Outcomes Statements

Many faculty members choose to incorporate words that reflect critical or higher-order thinking into their learning outcome statements. Bloom (1956) developed a taxonomy outlining the different types of thinking skills people use in the learning process. Bloom

argued that people use different levels of thinking skills to process different types of information and situations. Some of these are basic cognitive skills (such as memorization) while others are complex skills (such as creating new ways to apply information). These skills are often referred to as *critical thinking skills* or *higher-order thinking skills*.

Bloom proposed the following taxonomy of thinking skills. All levels of Bloom's taxonomy of thinking skills can be incorporated into expected learning outcome statements. Recently, Anderson and Krathwohl (2001) adapted Bloom's model to include language that is oriented towards the language used in expected learning outcome statements. A summary of Anderson and Krathwohl's revised version of Bloom's taxonomy of critical thinking is provided below.

Definitions of the different levels of thinking skills in Bloom's taxonomy

1. **Remember** – recalling relevant terminology, specific facts, or different procedures related to information and/or course topics. At this level, a student can remember something, but may not really understand it.

2. **Understand** – the ability to grasp the meaning of information (facts, definitions, concepts, etc.) that has been presented.

3. **Apply** – being able to use previously learned information in different situations or in problem solving.

4. **Analyze** – the ability to break information down into its component parts. Analysis also refers to the process of examining information in order to make conclusions regarding cause and effect, interpreting motives, making inferences, or finding evidence to support statements/arguments.

5. **Evaluate** – being able to judge the value of information and/or sources of information based on personal values or opinions.

6. **Create** – the ability to creatively or uniquely apply prior knowledge and/or skills to produce new and original thoughts, ideas, processes, etc. At this level, students are involved in creating their own thoughts and ideas.

List of Action Words Related to Critical Thinking Skills

Figure 3 shows a list of action words that can be used when creating the expected student learning outcomes related to critical thinking skills in a course. These terms are organized according to the different levels of higher-order thinking skills contained in Anderson and Krathwohl's (2001) revised version of Bloom's taxonomy.

4. TIPS FOR DEVELOPING COURSE LEVEL EXPECTED LEARNING OUTCOMES STATEMENTS

Limit the course-level expected learning outcomes to 5 - 10 statements for the entire course (more detailed outcomes can be developed for individual units, assignments, chapters, etc.).

Focus on overarching or general knowledge and/or skills (rather than small or trivial details).

Focus on knowledge and skills that are central to the course topic and/or discipline.

Create statements that are student-centered rather than faculty-centered (e.g., "upon completion of this course students will be able to list the names of all Data Mining techniques " versus "one objective of this course is to teach the names of all Data Mining techniques").

REMEMBER	UNDERSTAND	APPLY	ANALYZE	EVALUATE	CREATE
Count	Associate	Add	Analyze	Appraise	Categorize
Define	Compute	Apply	Arrange	Assess	Combine
Describe	Convert	Calculate	Breakdown	Compare	Compile
Draw	Defend	Change	Combine	Conclude	Compose
Identify	Discuss	Classify	Design	Contrast	Create
Label	Distinguish	Complete	Detect	Criticize	Drive
List	Estimate	Compute	Develop	Critique	Design
Match	Explain	Demonstrate	Diagram	Determine	Devise
Name	Extend	Discover	Differentiate	Grade	Explain
Outline	Extrapolate	Divide	Discriminate	Interpret	Generate
Point	Generalize	Examine	Illustrate	Judge	Group
Quote	Give examples	Graph	Infer	Justify	Integrate
Read	Infer	Interpolate	Outline	Measure	Modify
Recall	Paraphrase	Manipulate	Point out	Rank	Order
Recite	Predict	Modify	Relate	Rate	Organize
Recognize	Rewrite	Operate	Select	Support	Plan
Record	Summarize	Prepare	Separate	Test	Prescribe
Repeat		Produce	Subdivide		Propose
Reproduce		Show	Utilize		Rearrange
Select		Solve			Reconstruct
State		Subtract			Related
Write		Translate			Reorganize
		Use			Revise
					Rewrite
					Summarize
					Transform
					Specify

Figure 3: List of Action Words (Ref: Revised Version of Bloom's Taxonomy)

Focus on the learning that *results* from the course rather than describing activities or lessons in the course.

Incorporate or reflect the institutional and departmental missions.

Incorporate various ways for students to show success (outlining, describing, modeling, depicting, etc.) rather than using a single statement such as "at the end of the course, students will know _____ " as the stem for each expected outcome statement.

5. SAMPLE COURSE LEARNING OUTCOMES

The following depict some sample expected learning outcomes statements from the selected courses.

Data Structures:
After completing this course the student must demonstrate the knowledge and ability to:

1. **Understand** the concept of recursion and describe its implementation using a stack.
2. **Compare** iterative and recursive solutions for elementary problems.
3. **Describe** the usage and operations for maintaining various data structures.
4. **Choose** the appropriate data structure for a specific application.
5. **Apply** the notations used to analyze the performance of algorithms.
6. **Describe** various data structures like Stacks, Queues, Linked lists, Trees and Graphs are represented in memory and used by algorithms.
7. **Write** programs that use various data structures like Stacks, Queues, Linked lists, Trees and Graphs.
8. **Compare and contrast** the time complexities of various searching and sorting algorithms.
9. **Design and implement** an appropriate hashing function for an application.
10. **Apply** tree and graph traversal methods in real time applications.
11. **Describe** the concept of recursion, give examples of its use, describe how it can be implemented using a stack.
12. **Apply** basic algorithm strategies and to design algorithms for concrete problems of reasonable difficulty.

Computer Networks:

After completing this course the student must demonstrate the knowledge and ability to:

1. **Understand** basic computer network technology.
2. **Understand** and explain Data Communications System and its components.
3. **Enumerate** the layers of the OSI model and TCP/IP. Explain the function(s) of each layer.
4. **Identify** the different types of network topologies and protocols.
5. **Identify** the shortest path in a given network.
6. **Model** mathematically various error control schemes.
7. **Analyze** different LLC multiplexing mechanisms, node-to-node flow and error control.
8. **Analyze** different MAC mechanisms (Aloha, Slotted Aloha, TDMA, FDMA) and understand their pros and cons.
9. **Identify** the different types of network devices and their functions within a network.
10. **Enable** to interconnect various heterogeneous networks.
11. **Understand** and building the skills of subnetting and routing mechanisms.
12. **Design** and implement a peer to peer file sharing application utilizing application layer protocols such as HTTP, DNS, and SMTP and transportation layer protocol.
13. **Predict** ethical, legal, security and social issues related to computer networks.

Linux Programming:

After completing this course the student must demonstrate the knowledge and ability to:

1. **Identify** and use Linux utilities to create and manage simple file processing operations, organize directory structures with appropriate security.
2. **Develop** shell scripts to perform more complex tasks.
3. **Illustrate** file processing operations such as standard I/O and formatted I/O.
4. **Generalize** Signal generation and handling signals.
5. **Develop** programs using different Inter Process Communication (IPC) Mechanisms.
6. **Use** multithreading concepts to reduce the wastage of CPU time.
7. **Design** various client server applications using TCP or UDP protocols.

Operating Systems:

After completing this course the student must demonstrate the knowledge and ability to:

1. **Understand** the difference between different types of modern operating systems, virtual machines and their structure of implementation and applications.
2. **Understand** the difference between process & thread, issues of scheduling of user-level processes/ threads and their issues.
3. **Produce** customized algorithmic solutions for given synchronization problems.
4. **Use** modern operating system calls and synchronization libraries in software/ hardware interfaces**.**
5. **Identify** the rationale behind various memory management techniques along with issues and challenges of main memory, virtual memory and file system.
6. **Infer** the performance of page replacement algorithms in various scenarios.
7. **Recognize** the issues related to file system interface and implementation, disk management.
8. **Compare and Contrast** the time complexities of various disk scheduling algorithms.
9. U**nderstand** the concepts of deadlock in operating systems and how they can be managed / avoided and implement them in multiprogramming system.

6. AN OVERVIEW OF ASSESSMENT

What is assessment?

According to Palomba and Banta (1999) assessment involves the systematic collection, review, and use of evidence or information related to student learning. Assessment helps faculty understand how well their students understand course topics/lessons. Assessment exercises are often anonymous. This anonymity allows students to respond freely, rather than trying to get the "right" answer or look good. Assessment exercises attempt to gauge students' understanding in order to see what areas need to be re-addressed in order to increase the students' learning.

In other words, assessment is the process of investigating (1) *what* students are learning and (2) *how well* they are learning it in relation to the stated *expected learning outcomes* for the

course. This process also involves providing feedback to the students about their learning and providing new learning opportunities/strategies to increase student learning.

For example, Dr. JVR initiates a class discussion on material from Chapter One and determines that most of the students are confused about Topic X. This class discussion served as a method for assessing student learning and helped determine the fact that student learning related to Topic X is somewhat lacking. Dr. JVR now has the opportunity to (1) inform the students that there is some confusion and (2) make adjustments to address this confusion (e.g., ask student to re-read Chapter One, re-lecture over Topic X, etc.). This assessment process helps increase students' learning.

What is the difference between "evaluation" and "assessment"?
Evaluation focuses on making a judgment about student work to be used in assigning marks that express the level of student performance. Evaluation is usually used in the process of *determining marks*. Evaluation typically occurs after student learning is assumed to have taken place (e.g., a final exam). Evaluation is part of the assessment process. Course assignments that are evaluated/graded (e.g., exams, papers, tutorials, etc.) are often seen as formal assessment techniques.

While evaluation is an important component of most classrooms, it does have some limitations. For example, if the class average on an exam is a 45%, is seems pretty clear that something went wrong along the way. When one has only evaluated the final learning product, it can be challenging to go back and discover what happened. It can also be difficult to address the situation or provide opportunities for students to learn from their mistakes. Yes, a curve on an exam can help address a low class average, but does it help the students learn? Engaging in informal assessment activities throughout the course can help avoid this situation.

What is involved in the assessment process?
1. Establishing *expected learning outcomes* for the course;
2. Systematically gathering, analyzing, and interpreting evidence *(through formal assessment activities such as exams or papers and informal assessment activities such as in-class discussions exercises)* to determine how well the students' learning matches:
    - Faculty expectations for what students will learn;
    - The stated expected learning outcomes for the course;
3. Faculty members should use this evidence/assessment of student learning to:
    - Provide questionnaire to students about their learning (or lack thereof);
    - Adjust their teaching methods and/or students' learning behaviors to ensure greater student learning (Maki, 2004);

The Best Practice in a Classroom Assessment and is an example of a method that can be used to assess learning outcomes. At the end of a class period or major topic, faculty ask students to anonymously write down what point(s) were the most unclear to them. After class, faculty members review these responses and then re-teach or re-address any confusing topics, thus increasing student learning (Angelo & Cross, 1993).

7. WRITING A COURSE PURPOSE

When planning a course and determining the Learning Outcomes for that course, it is important to examine the course's purpose within the context of the college, and/or the department/program. This process will assist faculty in determining the intent of the course as well as how the course fits into the curriculum. This will help identify the essential knowledge, skills, etc. that should be incorporated into the course and the stated expected

learning outcomes for the course. The course purpose section should clarify the course's standing within the programme *(e.g., is the course required or an elective?, does this class have a pre-requisite?, etc.)*. It should also describe the course's role in the departmental/programmatic curriculum by addressing the intent (importance, main contribution, intrinsic value, etc.) of the class.

STEP ONE: Determine if the course is part of the IEEE / ACM / AICTE Model Curriculum

The earliest curriculum was published in 1968 for computer science (CS) by the Association for Computing Machinery (ACM), and in 1977 the Computer Society of the Institute for Electrical and Electronic Engineers (IEEE-CS) provided its first curriculum recommendations. In the late 1980's the ACM and the IEEE-CS together formed a task force to create curricula for computer science and computer engineering. The core curriculum covers classes in computer science curriculum, and subsequently separate curricula reports were issued for information systems, software engineering and computer engineering.

STEP TWO: Determine how the course fits into the departmental curriculum

Here are some questions to ask to help determine how a course fits in the departmental curriculum:

What role does the course play in the departmental/programmatic curriculum?

- Is this course required?
- Is this course an elective?
- Is this course required for some students and an elective for others?
- Does this class have a pre-requisite?
- Is this class a pre-requisite for another class in the department?
- Is this course part of IEEE / ACM / AICTE Model Curriculum?

How advanced is this course?

- Is this course an undergraduate or graduate course?
- Where does this course fall in students' degree plan - as an introductory course or an advanced course?
- Can I expect the students taking this course to know anything about the course topic?
- Are other faculty members counting on students who have taken this course to have mastered certain knowledge or skills?

When students leave this course, what do they need to know or be able to do?

- Is there specific knowledge that the students will need to know in the future?
- Are there certain practical or professional skills that students will need to apply in the future?
- Five years from now, what do you hope students will remember from this course?

What is it about this course that makes it unique or special?

- Why does the program or department offer this course?
- Why can't this course be "covered" as a sub-section of another course?
- What unique contributions to students' learning experience does this course make?
- What is the value of taking this course? How exactly does it enrich the program or department?

8. WRITING EXPECTED LEARNING OUTCOMES FOR A COURSE

The following pages should be of assistance in developing several broad, effectively stated expected learning outcomes for a course. When beginning to construct expected learning outcome statements, it is always good to think about the learners.

Please take a moment to think about the student learners in the course. Please consider the following questions:

- What are the most essential things the students need to know or be able to do at the end of this course?
- What knowledge and skills will they bring with them?
- What knowledge and skills should they learn from the course?

When you begin thinking about the expected learning outcomes for a course, it is a good idea to think broadly. Course-level expected learning outcomes do not need to focus on small details; rather, they address entire classes of theories, skill sets, topics, etc.

The "Course Description" contains the following contents: **(Annexure - A)**

- Course Overview
- Prerequisite(s)
- Marks Distribution
- Evaluation Scheme
- Course Objectives
- Course Outcomes
- How Program Outcomes are assessed
- How Program Specific Outcomes are assessed
- Syllabus
- List of Text Books / References / Websites / Journals / Others
- Course Plan
- Mapping course objectives leading to the achievement of the program outcomes and program specific outcomes
- Mapping course outcomes leading to the achievement of the program outcomes and program specific outcomes

9. REFERENCES

1. American Association of Law Libraries (2005). *Writing learning outcomes*.
2. Retrieved May 31, 2005 from http://www.aallnet.org/prodev/outcomes.asp .
3. Anderson, L.W., and Krathwohl, D.R. (Eds.) (2001). A taxonomy of learning, teaching, and assessment: A revision of Bloom's taxonomy of educational objectives. New York: Longman.
4. Angelo, T.A. & Cross, K.P. (1993). Classroom assessment techniques: A handbook for college teachers (2nd Ed.). San Francisco, CA: Jossey-Bass. Ball State University, (1999).
5. *Bloom's Classification of Cognitive Skills*. Retrieved
6. June 10, 2005 from http://web.bsu.edu/IRAA/AA/WB/chapter2.htm .
7. Bloom, B.S., (1956) Taxonomy of educational objectives: The classification of educational goals: Handbook I, cognitive domain. Longmans, Green: New York, NY.
8. Hales, L.W. & Marshall, J.C. (2004). Developing effective assessments to improve teaching and learning. Norwood, MA: Christopher-Gordon Publishers, Inc.
9. Huba, M.E., (2005). Formulating intended learning outcomes. Retrieved June 16, 2005

10. Fromhttp://www.viterbo.edu/academic/titleiii/events/files/Jun04/Intended%20Learning %20Outcomes.ppt#256,1,Formulating Intended Learning Outcomes .
11. Kansas State University, (2004). *Assessment of student learning plan*. Retrieved
12. May 15, 2005 from http://www.k-state.edu/assessment/Library/templatew.doc.
13. Kansas State University, (2004). *Form for identifying strategies and processes for*
14. *the assessment of student learning outcome(s)*. Retrieved May 15, 2005 from http://www.k-state.edu/assessment/Library/strategies.pdf .
15. Kansas State University, (2005). *How to write student learning outcomes: Action*
16. *verb List – suggested verbs to use in each level of thinking skills*. Retrieved May 15, 2005 from http://www.k-state.edu/assessment/Learning/action.htm.
17. Krumme, G (2001). *Major categories in the taxonomy of educational objectives*
18. *(Bloom 1956)*. Retrieved June 6, 2005 from http://faculty.washington.edu /krumme/guides/bloom1.html .
19. Maki, P.L. (2004). Assessing for learning: Building a sustainable commitment across the institution. Stylus: Sterling, VA.
20. Palomba, C.A. & Banta, T.W. Eds. (2001). Assessing student competence in accredited disciplines: Pioneering approaches to assessment in higher education. Stylus: Sterling, VA.
21. Siebold, R. & Beal, M. (May 2005). Online course development guide: The workbook. Presented at The Teaching Professor Conference in Shaumburg, IL.
22. Suskie, L. (ed) (2001). Assessment to promote deep learning: Insight from AAHE's 2000 and 1999 Assessment Conferences.
23. Suskie, L. (2004). Assessing student learning: A common sense guide. Anker Publishing Company: Bolton, MA.
24. St. Edward's University Center for Teaching Excellence (2004). *Task Oriented Question*
25. *Construction Wheel Based on Bloom's Taxonomy*. Retrieved on May 17, 2005 from http://www.stedwards.edu/cte/resources/bwheel.htm.
26. Texas Tech University (2005). Texas Tech University 2005-06 Undergraduate and Graduate Catalog Volume LXXXII. Published by the Office of Official Publications: Lubbock.
27. TX. Texas Tech University Office of the Ombudsman, (2005). Syllabus Guide for Faculty: Tips for creating a conflict free syllabus. Retrieved June 9, 2005 from
28. http:// www.depts.ttu.edu/ombudsman/publications/SyllabusGuideforFaculty.doc.

ANNEXURE - A: SAMPLE COURSE DESCRIPTION FORM

INSTITUTE OF AERONAUTICAL ENGINEERING

Dundigal, Hyderabad - 500 043

COMPUTER SCIENCE AND ENGINEERING

COURSE DESCRIPTION FORM

Course Title	**DATA STRUCTURES**			
Course Code	**A30502**			
Regulation	**R13 - JNTUH**			
Course Structure	Lectures	Tutorials	Practicals	Credits
	4	-	-	4
Course Coordinator	Ms. B Padmaja, Associate Professor			
Team of Instructors	Mr. Ch Suresh Kumar Raju, Associate Professor Mr. A Harekrishna, Assistant Professor			

I. COURSE OVERVIEW:

Data Structures is a subject of primary importance to the discipline of Computer Science and Engineering. It is a logical and mathematical model of sorting and organizing data in a particular way in a computer, required for designing and implementing efficient algorithms and program development. Different kinds of data structures like arrays, linked lists, stacks, queues, etc. are suited to different kinds of applications. Some specific data structures are essential ingredients of many efficient algorithms, and make possible the management of huge amounts of data, such as large databases and internet indexing services. Nowadays, various programming languages like C, C++ and Java are used to implement the concepts of Data Structures, of which C remains the language of choice for programmers across the world.

II. PREREQUISITE(S):

Level	Credits	Periods/ Week	Prerequisites
UG	4	4	Computer Programming

III. MARKS DISTRIBUTION:

Sessional Marks	University End Exam marks	Total marks
Midterm Test There shall be two midterm examinations. Each midterm examination consists of essay paper, objective paper and assignment. The essay paper is for 10 marks of 60 minutes duration and shall contain 4 questions. The student has to answer 2 questions, each carrying 5 marks. The objective paper is for 10 marks of 20 minutes duration. It consists of 10 multiple choice and 10 fill-in-the blank questions, the student has to answer all the questions and each carries half mark.	75	100

Sessional Marks	University End Exam marks	Total marks
First midterm examination shall be conducted for the first two and half units of syllabus and second midterm examination shall be conducted for the remaining portion. Five marks are earmarked for assignments. There shall be two assignments in every theory course. Assignments are usually issued at the time of commencement of the semester. These are of problem solving in nature with critical thinking. Marks shall be awarded considering the average of two midterm tests in each course.		

IV. EVALUATION SCHEME:

S. No.	Component	Duration	Marks
1	I Mid Examination	80 minutes	20
2	I Assignment	-	5
3	II Mid Examination	80 minutes	20
4	II Assignment	-	5
5	External Examination	3 hours	75

V. COURSE OBJECTIVES:

At the end of the course, the students will be able to:

I. Be familiar with basic techniques of algorithm analysis.

II. Be familiar with writing recursive methods.

III. Master the implementation of linked data structures such as linked lists, binary trees and graphs.

IV. Be familiar with binary tree traversal algorithms such as inorder, preorder and postorder and graph traversal and search algorithms.

V. Be familiar with advanced data structures such as balanced search trees, hash tables and priority queues.

VI. Be familiar with several searching and sorting algorithms including quick sort, merge sort and heap sort.

VII. Master analyzing and writing program solutions to problems using the above techniques.

VI. COURSE OUTCOMES:

After completing this course the student must demonstrate the knowledge and ability to:

1. **Understand** the concept of recursion and describe its implementation using a stack.
2. **Compare** iterative and recursive solutions for elementary problems.
3. **Describe** the usage and operations for maintaining various data structures.
4. **Choose** the appropriate data structure for a specific application.
5. **Apply** the notations used to analyze the performance of algorithms.
6. **Describe** various data structures like Stacks, Queues, Linked lists, Trees and Graphs are represented in memory and used by algorithms.

7. **Write** programs that use various data structures like Stacks, Queues, Linked lists, Trees and Graphs.
8. **Compare and contrast** the time complexities of various searching and sorting algorithms.
9. **Design and implement** an appropriate hashing function for an application.
10. **Apply** tree and graph traversal methods in real time applications.
11. **Describe** the concept of recursion, give examples of its use, describe how it can be implemented using a stack.
12. **Apply** basic algorithm strategies and to design algorithms for concrete problems of reasonable difficulty.

VII. HOW PROGRAM OUTCOMES ARE ASSESSED:

Program Outcomes		Level	Proficiency assessed by
PO1	**Engineering knowledge**: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.	H	Assignments, Tutorials
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.	H	Assignments
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.	H	Mini Projects
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.	S	Projects
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.	N	--
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.	N	--
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.	N	--
PO8	**Ethics**: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.	N	--
PO9	**Individual and team work**: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.	N	--
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,	N	--

Program Outcomes		Level	Proficiency assessed by
	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.	N	--
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.	S	Projects

N - None **S - Supportive** **H - Highly Related**

VIII. HOW PROGRAM SPECIFIC OUTCOMES ARE ASSESSED:

Program Specific Outcomes		Level	Proficiency assessed by
PSO1	**Professional Skills:** The ability to understand, analyze and develop computer programs in the areas related to algorithms, system software, multimedia, web design, big data analytics, and networking for efficient design of computer-based systems of varying complexity.	H	Lectures, Assignments
PSO2	**Problem-Solving Skills:** The ability to apply standard practices and strategies in software project development using open-ended programming environments to deliver a quality product for business success.	H	Projects
PSO3	**Successful Career and Entrepreneurship:** The ability to employ modern computer languages, environments, and platforms in creating innovative career paths, to be an entrepreneur, and a zest for higher studies.	S	Guest Lectures

N - None **S - Supportive** **H - Highly Related**

IX. SYLLABUS:

UNIT – I

Basic concepts- Algorithm Specification-Introduction, 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.

Singly Linked Lists-Operations-Insertion, Deletion, Concatenating singly linked lists, Circularly linked lists-Operations for Circularly linked lists, Doubly Linked Lists- Operations- Insertion, Deletion. Representation of single, two dimensional arrays, sparse matrices-array and linked representations.

UNIT – II

Stacks and Queue- Stack ADT, definition, operations, array and linked implementations in C, applications-infix to postfix conversion, Postfix expression evaluation, recursion implementation, Queue ADT, definition and operations ,array and linked Implementations in C, Circular queues-Insertion and deletion operations, Dequeue (Double ended queue)ADT, array and linked implementations in C.

UNIT – III

Trees – Terminology, Representation of Trees, Binary tree ADT, Properties of Binary Trees, Binary Tree Representations-array and linked representations, Binary tree traversals, Threaded binary trees,

Max Priority Queue ADT-implementation-Max Heap-Definition, Insertion into a Max Heap, Deletion from a Max Heap.

Graphs – Introduction, Definition, Terminology, Graph ADT, Graph Representations- Adjacency matrix, Adjacency lists, Graph traversals- DFS and BFS.

UNIT – IV

Searching- Linear Search, Binary Search, Static Hashing-Introduction, hash tables, hash functions, verflow Handling.

Sorting-Insertion Sort, Selection Sort, Radix Sort, Quick sort, Heap Sort, Comparison of Sorting methods.

UNIT – V

Search Trees-Binary Search Trees, Definition, Operations- Searching, Insertion and Deletion, AVL Trees-Definition and Examples, Insertion into an AVL Tree ,B-Trees, Definition, B-Tree of order m, operations-Insertion and Searching, Introduction to Red-Black and Splay Trees(Elementary treatment-only Definitions and Examples), Comparison of Search Trees.

Pattern matching algorithm- The Knuth-Morris-Pratt algorithm, Tries (examples only).

TEXT BOOKS:

1. Anderson-Freed, Susan, Ellis Horowitz, and Sartaj Sahni, "Fundamentals of Data Structures in C", Universities Press, 2e, 1992.
2. D. S. Kushwaha and A. K. Misra, “Data structures-A Programming Approach with C”, PHI Learning, 2012.

REFERENCES:

1. R. F. Gilberg And B. A. Forouzan, “Data structures: A Pseudocode Approach with C”, 2e, Cengage Learning.
2. M. A. Weiss, “Data structures and Algorithm Analysis in C”, 2e, Pearson.
3. A. M. Tanenbaum,Y. Langsam, M. J. Augenstein, “Data Structures using C”, Pearson.
4. R. Kruse, C. L. Tondo and B. Leung, “Data structures and Program Design in C”, 2e, Pearson.
5. R. Thareja, “Data Structures using C”, Oxford University Press.
6. S. Lipscutz, “Data Structures”, Schaum’s Outlines, TMH.
7. A. K. Sharma, “Data structures using C”, 2e, Pearson.
8. D. Samantha, “Classic Data Structures”, 2e, PHI.

X. COURSE PLAN:

At the end of the course, the students are able to achieve the following course learning outcomes:

Lecture No.	Topics to be covered	Course Learning Outcomes	Reference
1 – 3	**Basic concepts**- Algorithm Specification-Introduction and Recursive algorithms.	**Differentiate** iterative and recursive algorithm development procedures. Develop algorithms using recursive principle.	T1: 1.3
4 – 6	Performance analysis- time complexity and space complexity. Asymptotic Notation-Big O, Omega and Theta notations.	**Analyze** the algorithm and determine the time and space complexity.	T1: 1.5

Lecture No.	Topics to be covered	Course Learning Outcomes	Reference
7	Introduction to Abstract Data Type, Linear and Non Linear data structures.	**Understand** the importance of structure and abstract data type, and their basic usability in different applications.	T1: 2.1
8 – 14	**Singly Linked Lists**-Operations-Insertion, Deletion, Concatenating singly linked lists. Circularly linked lists-Operations for Circularly linked lists. Doubly Linked Lists-Operations- Insertion, Deletion.	**Understand** basic operations of single, double and circular linked lists.	T1: 4.1, 4.2, 4.5, 4.8
15 – 16	Representation of single, two dimensional arrays, sparse matrices-array and linked representations.	**Apply** the concept of arrays and linked list in problem solving.	T1: 2.1, 2.2, 2.5, 4.7
17 – 19	**Stacks and Queue**- Stack ADT, definition, operations, array and linked implementations in C.	**Identify** basic operations of Stack and its implementation.	T1: 3.1, 3.2, 4.3
20 – 23	Applications-infix to postfix conversion, Postfix expression evaluation, recursion implementation.	**Apply** Stack concepts in algebraic expression conversion.	T1: 3.6
24 – 25	Queue ADT, definition and operations ,array and linked Implementations in C.	**Identify** basic operations of Queue and its implementation.	T1: 3.3, 4.3
26 – 28	Circular queues-Insertion and deletion operations, Deque (Double ended queue)ADT, array and linked implementations in C.	**Identify** basic operations of Circular Queue, DEQUE and its implementation.	T1: 3.4, 4.3
29 – 32	**Trees** – Terminology, Representation of Trees, Binary tree ADT. Properties of Binary Trees, Binary Tree Representations-array and linked representations.	**Understand** basic operations of Tree and its representation.	T1: 5.1, 5.2
33 – 35	Binary tree traversals, Threaded binary trees.	**Construct** and traverse the binary tree.	T1: 5.3, 5.5
36 – 39	Max Priority Queue ADT-implementation-Max Heap-Definition, Insertion into a Max Heap, Deletion from a Max Heap.	**Understand** and Construct Priority Queue and Max Heap.	T1: 5.6
40 – 43	**Graphs** – Introduction, Definition, Terminology, Graph ADT. Graph Representations- Adjacency matrix, Adjacency lists, Graph traversals- DFS and BFS.	**Understand** the graph basics, its representation and traversal and search.	T1: 6.1, 6.2
44 – 45	**Searching**- Linear Search and Binary Search.	**Apply** searching techniques in different scenarios.	T1: 1.3
46 - 48	Static Hashing-Introduction, hash tables, hash functions, Overflow Handling.	**Understand** the need of hashing for fast retrieval of data.	T1: 8.2
49 – 53	**Sorting**-Insertion Sort, Selection Sort, Radix Sort, Quick sort, Heap Sort, Comparison of Sorting methods.	**Use** various kinds of sorting techniques and know when to choose which technique.	T1: 1.3, 7.2, 7.3, 7.6, 7.9

Lecture No.	Topics to be covered	Course Learning Outcomes	Reference
54 – 56	**Search Trees**-Binary Search Trees, Definition, Operations- Searching, Insertion and Deletion. **AVL Trees**-Definition and Examples, Insertion into an AVL Tree.	**Understand** and summarize the operations of binary search trees and AVL trees.	T1: 5.7, 10.2
57 – 58	B-Trees, Definition, B-Tree of order m, operations-Insertion and Searching.	**Understand** need of height balanced trees and construct B-Tree of order m.	T1: 11.1, 11.2
59 – 60	Introduction to Red-Black and Splay Trees (Elementary treatment-only Definitions and Examples), Comparison of Search Trees.	**Compare** different types of search trees.	T1: 10.3, 10.4
61 - 62	**Pattern matching algorithm**- The Knuth-Morris-Pratt algorithm, Tries.	**Understand** basics pattern matching algorithm.	T1: 2.7

XI. MAPPING COURSE OBJECTIVES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Objectives	Program Outcomes												Program Specific Outcomes		
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
I	H	H										S	H	S	
II	S	H	H										H	S	
III		H	S	S									S	H	
IV	H	S											H	S	
V					S								H		S
VI		H	H		S								H	S	S
VII	S	S	H									S	H	H	S

S – Supportive **H - Highly Related**

XII. MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

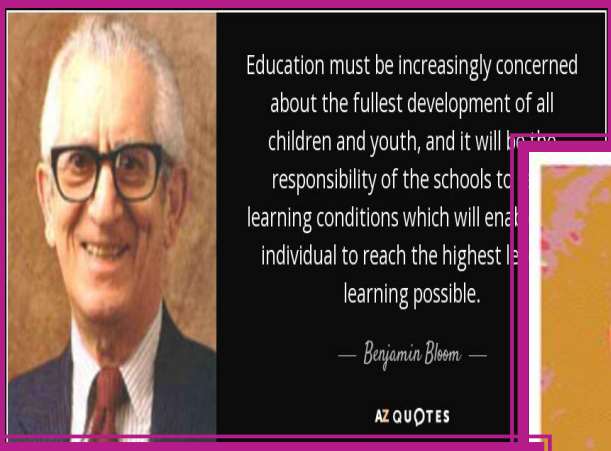
Course Outcomes	Program Outcomes												Program Specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
1	H	S	S										H	S	
2	H			S									S	H	
3			H		S								H	S	
4	S	H											S	H	
5	H	S											S	H	

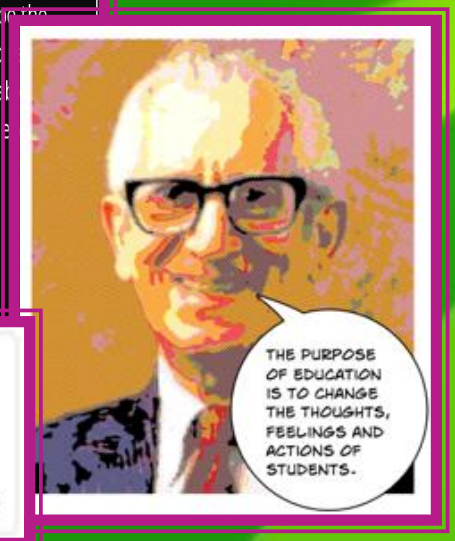
6	H			S								S	H	S	
7	S			H									S	H	
8	S	H											H	S	
9			H	H	S							S	S	H	
10	H			S									S	H	S
11	H			S	S								H	S	
12	H		H									S	S	H	S

S – Supportive **H - Highly Related**

Prepared by **:** Ms B. Padmaja, Associate Professor, CSE

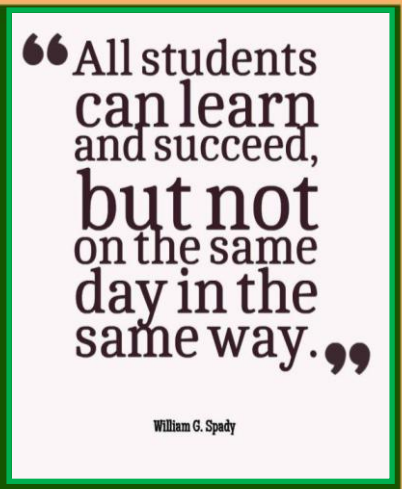
HOD, COMPUTER SCIENCE AND ENGINEERING





OUTCOMES

* *Planning begins with exit outcomes and moves to the creation of objectives*
* *Outcomes are a culminating demonstration of learning*
* *Restructuring of curriculum must move from "Traditional" to "Transitional" and "Transformational"*



Traditional Education vs. OBE

* *Textbook-centered vs. learner-centered*
* *Specific requirements vs. broad-goals*
* *Teach and proceed vs. teach for mastery*
* *Time-based vs. flexibility for mastery*