Programming for Problem Solving

As per New JNTU Syllabus

(Common to All branches for B.Tech First Year)

Authors

Dr. S Viswanadha Raju
Professor & Vice Principal
Department of CSE
JNTUH College of Engineering Jagtial
Jagtial Dist- 505 501, TS

Dr. L.V. Narasimha Prasad
Professor & Principal
Department of CSE
Institute of Aeronautical Engineering
Dundigal, Hyderabad-500 043, TS

Dr. K Venkateswara Reddy
Professor & Principal
Department of CSE
Marri Laxman Reddy Institute of Technology and
Management, Dundigal, Hyderabad – 500 043, TS

Dr. R. Madana Mohana
Professor
Department of CSE
Bharat Institute of Engineering and Technology
Ibrahimpatnam, Hyderabad- 501 510, TS

Dr. Nagalakshmi
Professor
Department of CSE
Marri Laxman Reddy Institute of Technology and Management
Dundigal, Hyderabad – 500 043, TS

Seven Hills
INTERNATIONAL PUBLISHERS
standard series...

All rights reserved. No part of this publication which is material protected by this copyright notice may be reproduced or transmitted or utilized or stored in any form or by any means now known or hereinafter invented, electronic, digital or mechanical, including photocopying, scanning, recording or by any information storage or retrieval system, without prior written permission from the ***Publisher.***

Information contained in this book has been published by **StudentsHelpline Publishing House (P) Ltd. , Hyderabad** and has been obtained by its Authors from sources believed to be reliable and are correct to the best of their knowledge. However, the Publisher and its Authors shall in no event be liable for any errors, omissions or damages arising out of use of this information and specifically disclaim any implied warranties or merchantability or fitness for any particular use.

COPYRIGHT REGISTRATION DIARY NUMBER**: 15367/2019-CO/L**

SevenHills International Publishers, Hyderabad

A Part of StudentsHelpline Publishing House (P) Ltd.

(An ISO 9001 : 2015 Certified Company)

Head Office

# 326/C, Level - 2, Surneni Nilayam
Near B K Guda Park, S R Nagar, Hyderabad - 500 038, INDIA
P.No:+91 40 23710657, 238000657 Fax: +91 40 23810657

Reg. Off

# 5-68, Pedda Gorpadu, Pakala, Tirupati, Chittoor - 517 112 AP, INDIA
mail:studentshelpline.in@gmail.com
www.**studentshelpline.**in

© SevenHills International Publishers

First Edition-2019

` 424/- Student Edition

` 624/- Library Edition with HB

Printed at StudentsHelpline Group, S R Nagar, Hyderabad-38
Published by Surneni Mohan Naidu for SevenHills International Publishers, Hyderabad - 38

Programming for Problem Solving

Course Objectives

- To learn the fundamentals of computers.
- To understand the various steps in program development.
- To learn the syntax and semantics of C programming language.
- To learn the usage of structured programming approach in solving problems.

Course Outcomes: The student will learn

- To write algorithms and to draw flowcharts for solving problems.
- To convert the algorithms/flowcharts to C programs.
- To code and test a given logic in C programming language.
- To decompose a problem into functions and to develop modular reusable code.
- To use arrays, pointers, strings and structures to write C programs.
- Searching and sorting problems.

Unit - 1: Introduction to Programming

Introduction to components of a computer system: disks, primary and secondary memory, processor, operating system, compilers, creating, compiling and executing a program etc.,

Number systems

Introduction to Algorithms: steps to solve logical and numerical problems. Representation of Algorithm, Flowchart/Pseudo code with examples, Program design and structured

programming

Introduction to C Programming Language: variables (with data types and space requirements), Syntax and Logical Errors in compilation, object and executable code , Operators, expressions and precedence, Expression evaluation, Storage classes (auto, extern, static and register), type conversion, The main method and command line arguments

Bitwise operations: Bitwise AND, OR, XOR and NOT operators

Conditional Branching and Loops: Writing and evaluation of conditionals and consequent branching with if, if-else, switch-case, ternary operator, goto, Iteration with for, while, do-while loops

I/O: Simple input and output with scanf and printf, formatted I/O, Introduction to stdin, stdout and stderr.

Command line arguments

Unit - II: Arrays, Strings, Structures and Pointers

Arrays: one and two dimensional arrays, creating, accessing and manipulating elements of arrays Strings: Introduction to strings, handling strings as array of characters, basic string functions available in C (strlen, strcat, strcpy, strstr etc.), arrays of strings Structures: Defining structures, initializing structures, unions, Array of structures

Pointers: Idea of pointers, Defining pointers, Pointers to Arrays and Structures, Use of Pointers in self-referential structures, usage of self referential structures in linked list (no implementation) Enumeration data type

Unit - III: Preprocessor and File handling in C

Preprocessor: Commonly used Preprocessor commands like include, define, undef, if, ifdef, ifndef Files: Text and Binary files, Creating and Reading and writing text and binary files, Appending data to existing files, Writing and reading structures using binary files, Random access using fseek, ftell and rewind functions.

Unit - IV: Function and Dynamic Memory Allocation

Functions: Designing structured programs, Declaring a function, Signature of a function, Parameters and return type of a function, passing parameters to functions, call by value, Passing arrays to functions, passing pointers to functions, idea of call by reference, Some C standard functions and libraries

Recursion: Simple programs, such as Finding Factorial, Fibonacci series etc., Limitations of Recursive functions

Dynamic memory allocation: Allocating and freeing memory, Allocating memory for arrays of different data types

Unit - V: Introduction to Algorithms

Algorithms for finding roots of a quadratic equations, finding minimum and maximum numbers of a given set, finding if a number is prime number, etc.

Basic searching in an array of elements (linear and binary search techniques),

Basic algorithms to sort array of elements (Bubble, Insertion and Selection sort algorithms), Basic concept of order of complexity through the example programs

Programming for Problem Solving

Chapter-1: Computer and Programming Fundamentals

1.0 Aims and objectives **1**

1.1 Introduction of Computers **1**

1.2 Classification of Computers **3**

1.3 Overview of a Computer **7**

1.3.1 Characteristics **7**

1.3.2 Applications **8**

1.3.3 Limitations **9**

1.4 Parts of a Computer **9**

1.5 Memory Hierarchy **16**

1.6 Introduction to OS **18**

1.6.1 Classification of Operating System **19**

1.6.2 Purpose of Operating System **19**

1.6.3 Functions of Operating System **20**

1.7 Operational Overview of a CPU **20**

1.8 GenerationS and Classification of Programming Languages **22**

1.8.1 Generations of Computers **22**

1.8.2 Differences between the 3GL and 4GLs **25**

1.8.3 Classification of Programming Languages **26**

1.8.4 Differences between High Level and Low Level Languages/ Machine level languages **29**

1.9 Creating and Running Programmes **29**

1.9.1 System Development Tools **30**

1.9.1.1 Comparison of Compiler and Interpreter **31**

1.9.2 Writing, Compiling and Running Your Program **31**

1.10 Number System **34**

1.10.1 Binary, Decimal, Hexadecimal and Octal Conversions **38**

1.11 Summary **40**

1.12 Key Terms **40**

1.13 Frequently Asked Questions 41

1.14 Exercises 43

1.15 Multiple choice questions 44

Chapter-2: Problem Solving and Algorithms

2.0 Aims and objectives 47

2.1 Introduction to Algorithms 47

2.2 Problem Solving 48

2.2.1 Steps Involved in Solving Logical and Numerical Problems 49

2.3 Characteristics of an Algorithm 49

2.4 Strategy for Designing Algorithms 51

2.5 Different Ways of Stating Algorithms 52

2.5.1 Step-form, Pseudo-code, Flowchart 52

2.6 Structured Programming Concept 52

2.7 Implementation of Algorithms 53

2.8 Analysis of Algorithms 54

2.8.1 Efficiency of Algorithm 54

2.9 Simple Examples of the Algorithm 56

2.10 Pseudo Code 57

2.10.1 Pseudo Code Rules 58

2.10.2 Advantages of Pseudo Codes 58

2.10.3 Limitations of Pseudo Codes 28

2.11 Flow Charts 59

2.11.1 Types of Flow Charts 60

2.11.2 Advantages of Flowcharts 61

2.11.3 Differences between Flowchart and Algorithm 61

2.11.4 Limitations of Flowcharts 62

2.11.5 Simple Examples of the Flowchart 62

2.12 PRogram Development steps 63

2.13 Tracing an Algorithm to Depict logic 65

2.14 Specification for Converting Algorithms into Programs 66

2.15 Additional ALgorithms and Flowcharts 67

2.16 Summary 78

2.17 Key Terms 78
2.18 Frequently Asked Questions 79
2.19 Exercises 80
2.20 Multiple Choice Questions 80

Chapter-3: Basics of 'C'

3.0 Aims and objectives 83
3.1 Introduction of 'C' 83
3.2 Characteristics of 'C' Language 84
3.3 'C' Features 85
3.4 'C' Limitations 85
3.5 Basic Structure of 'C' Language 86
3.6 Writing the First C Program, Comments 89
3.6.1 Compiling and Executing C Programs 90
3.6.2 Syntax and Logical Errors in Compilation 91
3.7 Program Statements 91
3.8 'C' Tokens 92
3.8.1 Keywords 93
3.8.2 Identifiers 93
3.8.3 Constants 94
3.8.4 Escape Sequences 95
3.8.5 Special Symbols 96
3.8.6 'C' Operators 96
3.8.7 Variables 96
3.8.7.1 Declaring Variables 99
3.8.7.2 Initializing Variables 99
3.8.7.3 Assigning Values to Variables 100
3.9 Basic Data Types in C 101
3.10 Operators 105
3.10.1 Arithmetic Operators 106
3.10.2 Relational Operators 108
3.10.3 Logical Operators 109
3.10.4 Assignement Operators 111

3.10.5 Increment and Decrement Operators 114

3.10.6 Conditional Operators 116

3.10.7 Bitwise Operators 117

3.10.8 Special Operators 122

3.11 Expressions and Evaluation 125

3.12 Precedence and Associativity 128

3.13 Type Conversions 132

3.14 Type Casting 133

3.15 Input and Output Functions 137

3.15.1 Formatted Input and Output Functions 139

3.15.2 Non-formatted Input and Output Functions 141

3.16 Additional Programs 143

3.17 Summary 146

3.18 Key Terms 147

3.19 Frequently Asked Questions 148

3.20 Exercise 152

3.21 Multiple Choice Questions 154

Chapter - 4: Control Statements

4.0 Aims and Objectives 161

4.1 Introduction 161

4.2 Statements 162

4.3 Decision Making (or) Conditional Statements 162

4.3.1 if, if-else, nested if, nested if-else and else-if Statements 162

4.3.1.1 if Statement 162

4.3.1.2 if-else Statement 164

4.3.1.3 Nested if Statements 165

4.3.1.4 else-if Statement 166

4.3.1.5 Dangling else Problem 168

4.3.1.6 nested if-else Statements 169

4.3.2 switch case Statements 170

4.4 Iterative Statements (or) Loop Statements 173

4.4.1 while Loop/while Statements 173

4.4.2 do-while Loop/do-while Statements **175**
4.4.3 for Loop/for Statements **176**
4.4.4 Use of Comma Operator in for Loop **178**
4.4.5 Differences for while, do-while and for loop **180**
4.5 Nested Loops **180**
4.6 Special Control Statement **184**
4.6.1 goto Statement **184**
4.6.2 break Statement **186**
4.6.3 continue Statement **188**
4.6.4 return Statement **189**
4.6.5 exit Statementq **191**
4.6.6 Difference between break and continue Statements **192**
4.6.7 null Statement **193**
4.7 Additional Programs **193**
4.8 Factoring Methods **195**
4.9 Summary **201**
4.10 Key Terms **201**
4.11 Frequently Asked Questions **201**
4.12 Exercise **203**
4.13 Multiple Choice Questions **204**

Chapter-5: Arrays and Strings

5.0 Aims and Objectives **211**
5.1 Introduction **211**
5.2 Arrays Concepts **212**
5.2.1 Declaration of Arrays **213**
5.2.2 Initialization of Arrays **216**
5.2.3 Accessing Array Elements **218**
5.2.4 Storing Array Elements **220**
5.3 Calculating the Length of the Array **221**
5.4 Using Arrays in C **222**
5.4.1 Performing Operations on Arrays **223**
5.4.2 Arrays Limitations **224**

5.5 Types of Arrays 225

5.5.1 Onedimensional Arrays 225

5.5.2 Twodimensional Arrays 226

5.5.3 Multidimensional Arrays 228

5.5.4 Comparison of Singledimensional and Multidimensional Arrays 230

5.6 Examples of Two Dimensional Arrays 231

5.6.1 Addition of Arrays 231

5.6.2 Subtraction of Arrays 232

5.6.3 Multiplication of Arrays 233

5.7 Strings 237

5.7.1 Introduction 237

5.7.2 Declaration of Strings 237

5.8 String Header or ‘C’ Library Functions for Strings 238

5.9 Null-terminated String 239

5.10 ‘C’ Strings 240

5.11 Handling Strings as Array of Characters 242

5.12 String Input/Output Functions 243

5.13 Arrays of Strings 246

5.14 String Handling/Manipulation Functions 248

5.14.1 header files in “ctype.h” 252

5.15 Miscellaneous String Functions 256

5.16 Additional Examples 257

5.17 Summary 264

5.18 Key Terms 264

5.19 Frequently Asked Questions 265

Chapter-6: Functions

6.0 Aims and objectives 279

6.1 Introduction 279

6.2 Concept of Function 280

6.2.1 Purpose of Function 281

6.3 Using Functions 281

6.3.1 Function Prototype Declaration 281

6.3.2 Function Definition 282
6.3.3 Function Calling 284
6.3.4 Designing Structured Programs 285
6.3.4.1 Defining and Accessing of Functions 285
6.3.5 Return Statement 286
6.3.6 Characteristics of Function 287
6.3.7 Advantages of Functions 287
6.4 Passing Arguments (or) Passing Parameters 288
6.4.1 Call-by-Value Vs Call-by-reference 289
6.4.2 Differences between Call-by-Value and Call-by-reference 291
6.4.3 Passing Variable Number of Arguments to a Function 294
6.5 Formal Parameters and Actual Parameters 296
6.6 Void Functions 297
6.7 Function Invocation and Function Execution 297
6.8 Nesting of Functions 298
6.9 User defined Functions 300
6.10 Standard/Library functions 306
6.10.1 Built–in Functions 309
6.11 Static Functions 311
6.12 Scope of Variables 313
6.12.1 Scope Rules 313
6.13 Storage Classes 316
6.13.1 Auto Storage Class 316
6.13.2 Extern Storage Class 316
6.13.3 Register Storage Class 317
6.13.4 Static Storage Class 317
6.13.5 Comparison of Different Storage Class Variables 318
6.14 Passing Arrays to Functions 319
6.15 Type Qualifiers 319
6.16 Inline Functions 320

6.17 Recursion 321

6.17.1 Recursive Functions 321

6.17.2 Advantages of Recursion 322

6.17.3 Limitations of Recursion 322

6.17.4 Types of Recursion 323

6.18 Towers of Hanoi 324

6.19 Recursion vs Iteration 328

6.20 User defined functions 328

6.20.1 Introduction 328

6.20.2 Need for User Defined Functions 329

6.20.3 Return Values and Their Types 329

6.20.4 Functions Returning Non-integers 330

6.21 Additional Programs 332

6.22 Summary 335

6.23 Key TermS 336

6.24 Frequently Asked Questions 336

6.25 Exercises 338

6.26 Multiple choice Questions 339

Chapter-7: Pointers

7.0 Aims and objectives 353

7.1 Introduction 353

7.2 Pointers 353

7.2.1 Declaring Pointer Variables 353

7.2.2 Assigning Pointers 355

7.2.3 Initialization of a Pointer 355

7.2.4 Accessing a Pointer's Contents 356

7.3 Address and Indirection Operator 357

7.4 Uses of Pointers 358

7.5 Disadvantages of Pointers 359

7.6 Arrays and Pointers 359

7.6.1 Relationship between Pointers and Arrays 360

7.6.2 Accessing Elements of Two-Dimensional Array 361

7.6.3 Pointer and Multidimensional Arrays 362

7.7 Array of Pointers 363

7.8 Dynamic Memory Allocation 364

7.9 Pointers to Pointers 370

7.10 Pointers to Void 372

7.11 Null Pointers 374

7.12 Pointers for inter function communication 374

7.13 Pointers and Strings 376

7.14 Dangling Pointer 377

7.15 Pointer to Array 377

7.16 Function and Pointers 378

7.16.1 Pointers to Functions 379

7.16.2 Character Pointers and Functions 379

7.17 Passing Arguments to Functions using Pointer 382

7.18 Difference between Array Name and Pointer 383

7.19 Additional Programs 383

7.20 Summary 386

7.21 Key Terms 386

7.22 Frequently Asked Questions 387

7.23 Exercises 387

7.24 Multiple choice Questions 388

Chapter-8: Structures and Unions

8.0 Aims and objectives 397

8.1 Introduction 397

8.2 Declaring a Structure and its Members 397

8.2.1 The Type Definition (typedef) 399

8.3 Initialization of a Structure 400

8.4 Accessing Members of a Structure 401

8.5 Assigning Values/Operations on Structures 402

8.6 Size of a Structure 403

8.6.1 Using Sizeof Operator 403

8.6.2 Without Using Sizeof Operator 404

8.7 Arr ay of Structures 404
8.8 Differences between Arrays and Structure 405
8.9 Nested Structures 405
8.10 Structures and Functions 406
8.11 Selfreferential Structures 409
8.11.1 Usage of Self Referential Structures in Linked List 410
8.12 Table Lookup 411
8.13 Passing Structures through Pointers 413
8.14 Structures Containing Arrays 415
8.15 Structures containing Pointers 417
8.16 Bit Fields 418
8.17 Introduction to Unions 419
8.17.1 Declaring a Union and its Members 419
8.18 Initialization of a Union 421
8.19 Accessing Members of Union 422
8.20 Arrays of Unions Variables 423
8.21 Unions inside Structures 423
8.22 Enumerated Data Types 424
8.23 Additional Programs 425
8.24 Summary 429
8.25 Key Terms 429
8.26 Frequently Asked Questions 429
8.27 Exercises 430
8.28 Multiple Choice Questions 431

Chapter-9: File Handling in C

9.0 Aims and objectives 437
9.1 Introduction 437
9.2 Using Files C 438
9.3 Types of Files 438
9.3.1 Differences between Text and Binary Files 439
9.4 File Structure 440
9.5 Streams 441

9.6 File Operations 441

9.6.1 Opening File 442

9.6.2 Reading File 444

9.6.3 Writing to a File 445

9.6.4 Closing Files 446

9.6.5 State of File 447

9.6.6 Appending Data to Existing Files 448

9.7 Overview of Functions 450

9.8 File Input/Output Functions (Standard Library Input/Output Functions for Files) 451

9.8.1 fread() Function 454

9.8.2 fwrite() Function 455

9.9 File Handling in C 456

9.10 File Status Functions 457

9.10.1 Error Handling during File Operations 457

9.11 Positioning Functions 458

9.12 Detecting the End-of-file 459

9.13 Accepting Command Line Arguments 461

9.14 Remove–Renaming a File–Creating a Temporary File 464

9.15 Files of Records, Random Access to Files of Records 466

9.16 Other File Management Functions 468

9.17 Concept of binary files 469

9.17.1 Reading and Writing from Binary Files 470

9.17.2 Writing and Reading Structures using Binary Files 470

9.18 Additional Programs 472

9.19 Summary 476

9.20 Exercise 476

9.21 Multiple choice Questions 477

Chapter-10: Searching and Sorting

10.0 aims and objectives 485

10.1 Introduction 485

10.2 Searching 486

10.2.1 Linear Search 486

10.2.2 Binary Search 490

10.2.2.1 Algorithm for Binary Search 491

10.2.2.2 Algorithm for Binary Search using Recursive Technique 493

10.2.3 Advantages and Disadvntages of Binary Search 495

10.2.4 Advantages and Disadvntages of Linear Search 496

10.3 Sorting 496

10.3.1 Bubble Sort 497

10.3.1.1 Advantages and Disadvntages of Bubble Sort 500

10.3.2 Selection Sort 500

10.3.2.1 Advantages and Disadvntages of Selection Sort 503

10.3.3 Insertion Sort 503

10.3.3.1 Advantages and Disadvntages of Selection Sort 507

10.3.4 Quick Sort 508

10.3.5 Comparison of Various Sorting Algorithms 512

10.4 Summary 514

10.5 Key Terms 515

10.6 Frequently Asked Questions 515

10.7 Exercises 515

10.8 Multiple Choice Questions 516

Chapter-11: Additional Features of C

11.0 Aims and objectives 519

11.1 Preprocessor Commands 519

11.2 Macros 525

11.3 Command-Line Arguments 531

11.4 Variable-length Argument Lists 534

11.5 Error Handling-stderr and Exit 534

11.6 Summary 535

11.7 Exercise 535