R22 B.Tech Syllabus
Programming For Problem Solving

B.Tech. I SEM

L	T	P	C
3	1	0	4

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 **10L**

Introduction to components of a computer system: disks, primary and secondary memory, processor, operating system, types of computer languages, compilers, creating, compiling and executing a program etc., Introduction to Algorithms: steps to solve logical and numerical problems. Representation of Algorithm, Flowchart with examples.

Introduction to C Programming Language: History, Basic Structure of a C program, variables (with data types and space requirements), Syntax and Logical Errors in compilation, object and executable code, Operators, expressions and precedence, Expression evaluation, type conversion, Bitwise operations: Bitwise AND, OR, XOR and NOT operators.

I/O: Simple input and output with scanf and printf.

UNIT - II: Loops ,Arrays, Strings, Structures (12L)

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

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 handling functions available in C .

Structures: Defining structures, initializing structures, Nested structures, Array of structures

Unions: Defining Unions,initializing unions,basic program on union.

Enumeration data type.

UNIT - III: Pointers , Pre processor and File handling in C

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).

Pre processor: Commonly used Pre processor 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 **,** Random access using fseek, ftell and rewind functions.

UNIT - IV: Function and Dynamic Memory Allocation(12L)

Functions: Designing structured programs, Declaring a function, Signature of a function, Parameters and return type of a function, categories of functions, passing parameters to functions, call by value, Passing arrays 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 , Storage classes (auto, extern, static and register), The main method and command line arguments.

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

UNIT - V: Introduction to Algorithms:(8L)

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).

TEXT BOOKS:

1. Byron Gottfried, Schaum'sOutline of Programming with C,McGraw-Hill
2. B.A. Forouzan and R.F. Gilberg C Programming and Data Structures, Cengage Learning, (3rdEdition)

REFERENCE BOOKS:

1. Brian W. Kernighan and Dennis M. Ritchie, The C Programming Language,Prentice Hall of India
2. R.G. Dromey, How to solve it by Computer, Pearson (16thImpression)
3. Programming in C, Stephen G. Kochan, Fourth Edition, Pearson Education.
4. Herbert Schildt, C: The Complete Reference, Mc Graw Hill, 4thEdition