18MC101COMPUTER PROGRAMMING AND PROBLEM SOLVING

Course Description and Objectives:

This course is aimed at offering fundamental concepts of programming language to the students. It starts with the basics of C-programming and deals with the structure and various attributes required for writing a ‘C’ program. It also introduces various operators and control statements used in programming. Then it switches to functions and arrays. It goes on with strings, pointers, files & the user defined data types. As a first-level course in computer science, it forms the basis to understand usage of various attributes in writing a program.

Course Outcomes:

The student will be able to:

- ➢ Analyze problems and develop solutions by writing algorithms.
- ➢ Design of various test cases for validating input/output data and functionality of the programs.
- ➢ Develop simple real-time applications to get familiarity of the programming environment.

Skills:

- Identify suitable data types for an application.
- Apply control statements for decision making problems.
- Use multidimensional array for matrix manipulations.
- Design a program to perform statistical analysis on given data.
- Analyze the difference between static & dynamic memory allocation.

Activities:

- Implementation of matrix operations.
- Implementation of string manipulation functions.
- Implementing dynamic memory allocation using malloc and calloc functions.
- Implementation of file operations.

Syllabus

UNIT – 1 **13 Hours**

PROGRAM STRUCTURE AND DATA TYPES: Structure of C program - Comments, Processor statement,Function header statement, Variable declaration statement and Executable statement; C character set, Constants, Identifiers, Operators, Punctuations, Keywords, Modifiers, Identifiers, Variables, C scopes, Basic data types, Typedef, Enumeration, Storage classes, Reading and writing characters, Formatted I/O.

UNIT - 2 **12 Hours**

OPERATORS AND CONTROL STATEMENTS: Operators - Assignment, Arithmetic, Relational,Logical, Bitwise, Ternary, Address, Indirection, Sizeof, Dot, Arrow, Parentheses operators; Expressions, Operator precedence, Associative rules, Control statements - Selection, Iteration, Jump, Label, Expression and Block.

UNIT - 3 **12 Hours**

FUNCTIONS AND ARRAYS: Function - Declaration, Prototype, Definition, Call by value and callby address, Standard library functions and Recursive functions; Array - Declaration, Initialization, Reading, Writing, Accessing and Passing as a parameter to functions, Multidimensional arrays.

UNIT - 4 **14 Hours**

STRINGS, POINTERS AND STRUCTURES: Strings - Declaration, String library functions, Array of strings, Commandline arguments; Pointers - Declaration, Initializing pointers, Multiple indirection, Relationship between arrays and pointers; Scaling up - Array of arrays, Array of pointers, Pointer to a pointer, Pointer to an array; Pointer to functions, Dynamic memory allocation functions.

Structures - Declaration, Initialization and accessing, Array of structures**,**Passing structures to functions, Structure pointers, Structures within structures; Unions, Bit-fields.

UNIT - 5 **10 Hours**

FILES, SORTING AND SEARCHING: Files - I/O and processing operations on text and binary files; Pre-processor directives, Sorting – Bubble sort, Selection sort, Insertion sort; Searching – Linear Search, Binary Search.

LIST OF EXPERIMENTS Total hours-30

1. Compute the factors of a number.
2. Compute the average of ‘n’ numbers.
3. Find whether a number is palindrome or not.
4. Find whether a number is a power of 2 or not.
5. Compute the factorial of a number.
6. Implement any kind of operation (+,-,*,/,%) using a switch case.
7. Swap two values using call by value and call by reference.
8. Using structure of arrays.
9. Find the reversal of a number.
10. Find the frequency of each number in the array.
11. Which takes 0’s & 1’s as input and the array should consist of all 0’s first and then 1’s.
12. Copy the first 10 words of a file into the other file.
13. Count the number of words in a file.
14. Create a structure which stores the student’s information in a class.
15. Reverse the contents of the array.
16. Implement pointer of pointers.
17. Give nth term of the Fibonacci number.
18. Find the factorial of a number using recursion.
19. Find the number of vowels in a file.

20. Access the structure and union members.
21. Program to arrange the given array of elements in ascending order using Bubble sort.
22. Program to arrange the given array of elements in ascending order using Selection sort.
23. Program to arrange the given array of elements in ascending order using Insertion sort.
24. Program to find an element from the given array of elements using Linear search.
25. Program to find an element from the given array of elements using binary search

Text Books:

1. Ajay Mittal, "Programming in C - A practical Approach", 1st edition, Pearson Education India, 2015.
2. ReemaThareja, "Introduction to C Programming", 2nd edition, Oxford University Press India, 2015.

Reference Books:

1. Herbert Schildt, C, "The Complete Reference", 4th edition, Tata McGraw-Hill, 2000.
2. E. Balagurusamy, "Programming in ANSI C", 4th edition, Tata McGraw- Hill, 2008.