Problem Solving through Programming

Subject code: 18ESCS01 **Total hours: 45**
Credits: 3 **Hours/week: 4**
L-T-P: 2-1-0

Module – 1

INTRODUCTION (8 Hours)

Generation and Classification of Computers- Basic Organization of a Computer –Number System – Binary – Decimal – Conversion – Problems. Need for logical analysis and thinking– Algorithm – Pseudo code – Flow Chart

Module – 2

C PROGRAMMING BASICS (10 Hours)

Problem formulation – Problem Solving - Introduction to ' C' programming –fundamentals – structure of a 'C' program – compilation and linking processes – Constants, Variables – Data Types – Expressions using operators in 'C' – Managing Input and Output operations – Decision Making and Branching – Looping statements – solving simple scientific and statistical problems.

Module – 3

ARRAYS AND STRINGS (9 Hours)

Arrays – Initialization – Declaration – One dimensional and Two dimensional arrays. String- String operations – String Arrays. Simple programs- sorting- searching – matrix operations.

Module – 4

FUNCTIONS AND POINTERS (9 Hours)

Function – definition of function – Declaration of function – Pass by value – Pass by reference – Recursion – Pointers - Definition – Initialization – Pointers arithmetic – Pointers and arrays- Example Problems.

Module – 5

STRUCTURES AND UNIONS (9 Hours)

Introduction – Need for structure data type – structure definition – Structure declaration – Structure within a structure - Union - Programs using structures and Unions – Storage classes, Pre-processor directives.

TEXTBOOKS:

1. Pradip Dey, Manas Ghosh, "Fundamentals of Computing and Programming in C", First Edition, Oxford University Press, 2009

2. Ashok N. Kamthane, "Computer programming", Pearson Education, 2007.

3. Yashavant P. Kanetkar. " Let Us C", BPB Publications, 2011.

REFERENCES:

1. Kernighan,B.W and Ritchie,D.M, "The C Programming language", Second Edition, Pearson Education, 2006

2. Byron S Gottfried, "Programming with C", Schaum's Outlines, Second Edition, Tata McGraw-Hill, 2006.

3. R.G. Dromey, "How to Solve it by Computer", Pearson Education, Fourth Reprint, 2007

Course Outcomes: At the end of the course, the student will be able to:

CO-1 Understand the components of computing systems, Develop algorithms for mathematical and scientific problems

CO-2 Choose data types and structures to solve mathematical and scientific problem

CO-3 Develop modular programs using control structures

CO-4 Write programs to solve real world problems using programming features