<div align="center">

II Semester
Course 3: Problem Solving using C
Credits -3

</div>

Course Objectives
1. To explore basic knowledge on computers
2. Learn how to solve common types of computing problems.
3. Learn to map problems to programming features of C.
4. Learn to write good portable C programs.

Course Outcomes
Upon successful completion of the course, a student will be able to:
1. Understand the working of a digital computer and Fundamental constructs of Programming
2. Analyze and develop a solution to a given problem with suitable control structures
3. Apply the derived data types in program solutions
4. Use the 'C' language constructs in the right way
5. Apply the Dynamic Memory Management for effective memory utilization

UNIT-I

Introduction to computer and programming: Introduction, Basic block diagram and functions of various components of computer, Concepts of Hardware and software, Types of software, Compiler and interpreter, Concepts of Machine level, Assembly level and high-level programming, Flowcharts and Algorithms

Fundamentals of C: History of C, Features of C, C Tokens-variables and keywords and identifiers, constants and Data types, Rules for constructing variable names, Operators, Structure of C program, Input /output statements in C-Formatted and Unformatted I/O

UNIT-II

Control statements: Decision making statements: if, if else, else if ladder, switch statements. Loop control statements: while loop, for loop and do-while loop. Jump Control statements: break, continue and goto.

UNIT-III

Derived data types in C: Arrays: One Dimensional arrays - Declaration, Initialization and Memory representation; Two Dimensional arrays -Declaration, Initialization and Memory representation.
Strings: Declaring & Initializing string variables; String handling functions, Character handling functions

UNIT-IV

Functions: Function Prototype, definition and calling. Return statement. Nesting of functions. Categories of functions. Recursion, Parameter Passing by address & by value. Local and Global variables. **Storage classes**: automatic, external, static and register.
Pointers: Pointer data type, Pointer declaration, initialization, accessing values using pointers. Pointer arithmetic. Pointers and arrays, pointers and functions.

UNIT-V

Dynamic Memory Management: Introduction, Functions-malloc, calloc, realloc, free **Structures:** Basics of structure, structure members, accessing structure members, nested structures, array of

structures, structure and functions, structures and pointers. **Unions** - Union definition; difference between Structures and Unions.

Text Books:
1. E. Balagurusamy, "Programming in ANSI C", Tata McGraw Hill, 6th Edn, ISBN-13: 978-1-25- 90046-2
2. Herbert Schildt, —Complete Reference with C, Tata McGraw Hill, 4th Edn., ISBN- 13: 9780070411838, 2000
3. Computer fundamentals and programming in C, REEMA THAREJA, OXFORD UNIVERSITY PRESS

Reference Books
1. E Balagurusamy, COMPUTING FUNDAMENTALS & C PROGRAMMING – Tata McGraw-Hill, Second Reprint 2008, ISBN 978-0-07-066909-3.
2. Ashok N Kamthane, Programming with ANSI and Turbo C, Pearson Edition Publ, 2002.
3. Henry Mullish&Huubert L.Cooper: The Spirit of C An Introduction to modern Programming, Jaico Pub. House,1996.
4. Y kanithkar, let us C BPB, 13th edition-2013, ISBN:978-8183331630,656 pages.

SUGGESTED CO-CURRICULAR ACTIVITIES & EVALUATION METHODS:
Unit 1: Activity: Quiz on computer hardware and software concepts
        Evaluation Method: Objective-based quiz assessing knowledge and understanding
Unit 2: Activity: Problem-solving using Decision-Making Statements
        Evaluation Method: Correctness of decision-making logic
Unit 3: Activity: Array and String Program Debugging
        Evaluation Method: Identification and correction of errors in code
Unit 4: Activity: Pair Programming Exercise on Functions
        Evaluation Method: Collaboration and Code Quality
Unit 5: Activity: Structured Programming Assignment
        Evaluation Method: Appropriate use of structures and nested structures