Course Type	Course Code	Name of Course	L	T	P	Credit
VAC2	NCSV102	Computer Programming Lab	0	0	2	1

Course Objective
The course will develop practical programming skills in the C programming language.
Learning Outcomes
● To become comfortable with using C development tools such as compilers, editors, and debuggers. ● Understanding basic syntax of C program ● To learn debugging techniques and strategies for identifying and fixing errors in C programs, and to execute test cases to validate program functionality. ● To solve programming challenges and exercises that reinforce understanding of concepts and encourage creative problem-solving. ● Understanding importance of clear and concise code documentation, commenting, and coding standards.

Unit No.	Topics to be Covered	Lecture Hours	Learning Outcome
1.	**Introduction to Programming:** Pseudocode and flowcharts, Overview of Programming Concepts, Introduction to the C Programming Language, Writing and Running Simple C Programs, Basic Input/Output Operations	2	Setting up the development environment and writing simple C programs
2.	**Data Types and Operators:** Basic data types in C, Constants and variable declaration, Operators & Expressions: Arithmetic, Relational, Logical, Bitwise operators, Conditional, Assignment, Order of evaluation and Type conversions	4	Programming in C to understand how to declare variable, using operations and building expressions.
3.	**Control Structures and Loops:** Control statements - if, else, nested if-else, switch, Loops - for, while, do-while, break and continue statements, Nested loops	4	Programming in C to understand how to use control and loops to solve problems
4.	**Arrays and strings:** Declaration, Initialization, Processing, Multi-dimensional Arrays. String Handling: Declaration, Initialization, Processing. I/O for string	4	Programming in C to understand how to use 1D and 2D arrays and string manipulation
5.	**Functions:** Defining functions, Accessing a function, Passing arguments, Scope and lifetime of variables, Recursive functions, Types of recursion, Storage classes	5	Basic understanding of functions, recursion, modularity of code using functions.
6.	**Pointers:** Declaration, Initialization, and Dereferencing, Pointer Arithmetic, Passing pointers to a function, Pointers and arrays, Dynamic Memory Allocation: malloc, calloc, realloc, free.	4	Programing using pointers, call by reference and dynamic memory allocations
7.	**User defined data types:** Structures, Structure and pointers, passing structure to a function, Self-referential structures, Unions, Enumerations	3	Programming in C and modeling real world data using structures
8.	**File Handling:** Opening, Reading, Writing, and Closing Files, File pointers and access modes, Command line arguments, Preprocessor directives	2	Implementing basic file manipulation programs in C

Text Books:

1. "Programming with C" by Byron Gottfried
2. "Let us C" by Yashavant Kanetkar

Reference Books:

1. Dennis Rictiche, The C Programming language
2. The Complete Reference C, by Herbert Schildt
3. "ANSI C" by E. Balagurusamy