Syllabus

Course Type	Course Code	Name of Course	L	T	P	Credit
	NCSV101	Computer Programming	3	0	0	9

Course Objective

The course will provide the basic concepts and develop programming skills in the C programming language.

Learning Outcomes

- Familiarize students with the fundamental concepts of programming
- Learning syntax and semantics of writing a valid C code
- Develop problem solving skills emphasizing algorithmic thinking
- Ability to translate or model real world problems into programmable code
- Ability to organizing code into separate modules/functions for code reusability and maintainability

Unit No.	Topics to be Covered	Lecture Hours	Learning Outcome
1.	**Introduction to Programming:** Fundamentals of Computing, History of computation, Pseudocode and flowcharts, Overview of Programming Concepts, Introduction to the C Programming Language, Writing and Running Simple C Programs, Basic Input/Output Operations	4	Familiarizing with how we can talk to computers using programming languages and developing the general idea of programming
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	5	Learning various data types available in C language and how to use them in various types of 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	6	Basic understanding of control and loops statement in c and programming
4.	**Arrays and strings:** Declaration, Initialization, Processing, Multi-dimensional Arrays. String Handling: Declaration, Initialization, Processing. I/O for string	6	Basic understanding of arrays, declaration and initialization and strings
5.	**Functions:** Defining functions, Accessing a function, Passing arguments, Scope and lifetime of variables, Recursive functions, Types of recursion, Storage classes	6	Learning the power of functions, modularization of code and use of recursion
6.	**Pointers:** Declaration, Initialization, and Dereferencing, Pointer Arithmetic, Passing pointers to a function, Pointers and arrays, Dynamic Memory Allocation: malloc, calloc, realloc, free.	6	Basic understanding of pointers and their usage in the development, Programming using 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	5	Learning user defined data types and how to model real data through them
8.	**File Handling:** Opening, Reading, Writing, and Closing Files, File pointers and access modes, Command line arguments, Preprocessor directives	4	Basic understanding of file management using C

Syllabus

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