C Programming – OA25CP01

Beginner to Expert Course

Course Description

Master the fundamentals of structured programming using C, the foundation for many modern languages. This course covers everything from syntax and data types to pointers, file handling, and data structures. Perfect for students, beginners, and professionals preparing for systems programming, embedded development, or competitive coding.

What You'll Learn

- Understand C syntax, variables, and operators
- Solve problems with structured programming
- Work with arrays, strings, pointers, and memory management
- Implement data structures in C
- Handle files and build real-world projects

Course Requirements

- No prior programming knowledge required
- A computer with GCC (MinGW/Clang) and an IDE (Code::Blocks/VS Code)

Syllabus (35 Sessions)

C Fundamentals

1. Installing GCC and Code::Blocks/VS Code
2. Introduction to C: structure of a program
3. Data types, variables, and constants
4. Operators: arithmetic, relational, logical, bitwise

Basic Programming Concepts

5. Input/output using scanf and printf
6. Control statements: if, if-else, nested if
7. Switch-case statements
8. Loops: for, while, do-while
9. Break, continue, and goto

Functions in C
10. Defining and calling functions
11. Parameters and return values
12. Recursion and its applications

Arrays and Strings
13. One-dimensional arrays

14. Multi-dimensional arrays (matrices)
15. String handling and library functions

Pointers and Memory Management
16. Basics of pointers
17. Pointer arithmetic and arrays
18. Dynamic memory allocation (malloc, calloc, free)
19. Pointers to functions

Structures and Unions
20. Defining and using structures
21. Nested structures and arrays of structures
22. Unions and enumerations

File Handling
23. File input and output functions
24. Sequential and random file access
25. Error handling in file operations

Data Structures in C
26. Implementing stacks and queues
27. Linked lists (singly, doubly, circular)
28. Searching and sorting algorithms

Advanced C Concepts
29. Preprocessor directives and macros
30. Header files and modular programming
31. Command line arguments

Practice Session
32. Mini Project 1: Student Record Management System

Projects
33. Project 2: Inventory Management using File Handling
34. Project 3: Banking System with Structures
35. Final Exam and Wrap-Up (certificate)