C++ Programming

Beginner to Expert Course

Course Description

This course introduces object-oriented programming with C++. Learn everything from C++ basics to advanced topics like templates, STL, file handling, and OOP design principles. Build real-world projects with OOP and master modern C++ (C++17/20).

What You'll Learn

- Understand C++ syntax, data types, and operators
- Master object-oriented programming (OOP) concepts
- Use inheritance, polymorphism, and templates
- Work with STL containers and algorithms
- Apply C++ to projects like management systems and simulations

Course Requirements

- Knowledge of basic programming (C helpful but not mandatory)
- A computer with GCC (g++), Code::Blocks, or VS Code

Syllabus (35 Sessions)

C++ Fundamentals

1. Installing g++ and IDE
2. C++ syntax vs. C syntax differences
3. Input/output with cin and cout
4. Data types, variables, and operators

Control Flow and Functions
5. Decision-making (if, switch)
6. Loops (for, while, do-while)
7. Functions, overloading, default arguments
8. Inline functions and recursion

Object-Oriented Programming Basics
9. Classes and objects
10. Constructors and destructors
11. Member functions and access specifiers
12. Static members and friend functions

Advanced OOP Concepts
13. Inheritance (single, multiple, multilevel)
14. Polymorphism (compile-time and run-time)
15. Virtual functions and abstract classes
16. Operator overloading

Templates and Exception Handling
17. Function templates
18. Class templates
19. Exception handling (try, catch, throw)

Standard Template Library (STL)
20. Working with vectors, lists, and deques
21. Maps, sets, and iterators
22. Algorithms in STL (sort, find, transform)

File Handling
23. File input and output streams
24. Reading/writing objects using files
25. Error handling in file I/O

Advanced C++ Features
26. Pointers to objects
27. Dynamic memory and smart pointers
28. Namespaces and inline namespaces
29. Lambda expressions (C++11+)

Practice Session
30. Mini Project 1: Library Management System

Projects
31. Project 2: Student Information System using OOP
32. Project 3: Banking System with Exception Handling
33. Project 4: Inventory Management using STL
34. Project 5: Online Quiz System
35. Final Exam and Wrap-Up (certificate)