

Module 1: Introduction to C++

- History and features of C++
- Difference between C and C++
- Structure of a C++ program
- Compilers and IDEs (Turbo C++, GCC, Visual Studio, Code::Blocks)
- Input/output operations (cin, cout)

Module 2: Basics of C++ Programming

- Data types, variables, constants
- Operators (arithmetic, relational, logical, bitwise)
- Control structures (if, switch, loops)
- Functions (declaration, definition, recursion)
- Arrays and strings

Module 3: Object-Oriented Programming (OOP)

- Concepts of OOP (encapsulation, inheritance, polymorphism, abstraction)
- Classes and objects
- Constructors and destructors
- Function overloading and operator overloading
- Inheritance (single, multiple, multilevel, hierarchical, hybrid)
- Virtual functions and dynamic polymorphism

Module 4: Advanced Features

- Pointers and memory management
- References and dynamic allocation (new, delete)
- File handling (read/write operations)
- Exception handling (try, catch, throw)
- Templates (function templates, class templates)
- Standard Template Library (STL) basics (vectors, lists, maps)

Module 5: Practical Applications

- Creating small programs (calculator, student record system)
- Implementing data structures (stack, queue, linked list)
- Mini project using OOP concepts (library management, inventory system)
- Final project: Complete application with file handling and OOP

Learning Outcomes

By the end of this course, students will:

- Understand C++ syntax and structure
- Apply OOP concepts to solve problems
- Use pointers, templates, and file handling effectively
- Build small to medium-sized applications in C++
- Gain confidence for advanced programming and competitive coding