

C++ Language

C++ Language is one of the approaches to provide object-oriented functionality with C like syntax. C++ adds greater typing strength, scoping and other tools useful in object oriented programming and permits generic programming via templates. It is regarded as a middle-level language, as it comprises a combination of both high-level and low-level language features. Some of its application domains include systems software, device drivers, embedded software, high-performance server and client applications, and entertainment software such as videogames.

Basics in C++

- Procedure Oriented Programming
- Principle of Object Oriented Programming
- Software evaluation
- Oop paradigm
- Basic concept of Oop
- Benefit of Oop
- Application of Oop
- Programming Methods
- Introduction to C++
- Tokens
- Keywords
- Identifiers
- Constants
- Operators
- Scope Resolution operator (::)
- Memory Management Operators
- Manipulators

Control Structures

- Sequential Control Structure
- Selective Control Structure
 - If, if...else, if....else if, nested if
 - Switch case statement
 - Nested Switch
- Repetitive Control Structure
 - For Loop
 - While Loop

- Do while Loop
- Continue and break
- Nested loop

Arrays and Strings

- Single dimensional array
- Two dimensional array
- Multi-dimensional array
- What are strings?
- String Manipulation Functions

Functions

- Introduction
- Defining a function
- Function Prototypes
- Return type in main Prototype
- Call by value and call by Reference
- Return by Reference
- Inline Function
- Friend Function
- Functions with default arguments
- Function Overloading
- Default arguments
- Actual and Formal Parameter
- Function overloading

Basic concepts of Object Oriented Programming

- Object
- Class
- Inheritance
- Polymorphism
- Data Abstraction
- Data Encapsulation
- Dynamic Binding
- Message Passing

Classes and Objects

- Introduction
- C structure Revisited
- Defining Classes in C++
- Class declaration
- Access specifiers
- Classes and Encapsulation
- Member Functions
- Instantiating and Using Classes
- Objects as arguments
- Difference between Class and Structure
- Returning objects
- Static member Data and Static Member Function
- Friend Functions and Friend classes

Constructors and Destructors

- Introduction
- Defining constructor
- Using Constructors
- Multiple Constructors and Initialization Lists
- Constructor overloading
- Default constructor
- Copy constructor
- Destructor
- Defining Destructor
- Using Destructors to Destroy Instances

Inheritance

- Overview of Inheritance ➤ What is Inheritance?
- Features or Advantages of Inheritance
- Private, Public, Protected Members
- Types of inheritance
- Single inheritance
- Multiple inheritances
- Multi-level inheritance

- Hierarchical Inheritance
- Hybrid Inheritance
- Defining Base and Derived Classes
- Constructor and Destructor Calls
- Inheritance scope

Pointers

- Reference Pointer(&)
- Dereference Pointer(*)
- Declaring variables of pointer types
- Pointer Initialization
- Pointers and Arrays
- Pointer Arithmetic
- Pointers to Functions
- Pointers to pointers
 - This pointer
 - Void pointer
 - Null pointer
 - Dynamic memory allocation
- New, delete operator

Operator Overloading

- Need of overloading
- Defining operator overloading
- Overloading unary operators
- Overloading binary operators
- Overloading binary operators using friend function
- Rules for operator overloading
- Overloading other operators

Polymorphism and Virtual function

- Types of polymorphism
- Early binding
- Dynamic binding
- Virtual Functions
- Pure Virtual function
- Null virtual function

Templates

- Introduction
- Templates
- Function Templates
- Class Templates
- Member function Templates
- Template Arguments

Exception Handling

- Introduction
- Syntax of Exception Handling Code
- Exception handling mechanism
- Try, catch, throw keywords

File Handling

- Introduction
- Classes for File Stream Operations
- Opening and closing a file
- If stream, Of stream, Stream
- Detecting End of file
 - Sequential Access files
 - Random Access files
 - Binary Files

Command line arguments