C and C++ Programming (For Freshers - Faculty Guide)

Created by Rajendra Sharma, SDE cum Technical Trainer

PART A: C Programming

1. Introduction to C Language

- Evolution of Programming Languages
- History of C Language (Dennis Ritchie, 1972)
- Features and Use Cases of C
- Structure of a C Program
- Compilation and Execution Flow (Preprocessor → Compiler → Assembler → Linker → Loader)
- Writing First Program: Hello World
- Setting up IDE (Turbo C, GCC, Code::Blocks, VS Code)

2. Variables, Constants, and Data Types

- What are Variables and Constants?
- Rules for Naming Variables (Identifiers)
- Keywords in C
- Data Types: int, float, char, double, void
- Type Qualifiers: short, long, signed, unsigned
- Format Specifiers: %d, %f, %c, %s, %u, etc.
- Input/Output using scanf() and printf()

3. Operators and Expressions

- Arithmetic Operators: +, -, *, /, %
- Relational Operators: ==, !=, >, <, >=, <=
- Logical Operators: &&, ||, !
- Bitwise Operators: &, |, ^, ~, <<, >>
- Assignment Operators: =, +=, -=, etc.
- Increment/Decrement: ++, --
- Ternary Operator: condition ? true : false
- Type Conversion (Implicit & Explicit Casting)

4. Decision Making and Looping (Control Statements)

- if, if-else, else-if Ladder
- switch-case with break
- **Loops:**
  - while loop
  - do-while loop
  - for loop
- **Loop Control:**
  - break, continue
  - goto (with example and caution)

5. Functions in C

- Function Declaration, Definition, and Call
- Arguments and Return Values
- Types of Functions: No Argument-No Return, With Argument-With Return
- Call by Value vs Call by Reference
- Recursion: Factorial, Fibonacci, Power, GCD, etc.

6. Arrays

- What is Array?
- Declaration, Initialization, and Traversal
- Single Dimensional Arrays
- Two Dimensional Arrays (Matrix Representation)
- Array Programs: Sum, Max/Min, Reverse, Sort
- Passing Arrays to Functions

7. Strings

- What is String? char[] Representation
- String I/O: gets(), puts(), scanf(), printf()
- String Functions: strlen(), strcpy(), strcat(), strcmp(), strrev()
- Manual Implementation of String Functions
- String and Pointer Relationship

8. Pointers

- Introduction to Pointers
- Declaration, Initialization, and Dereferencing
- Pointer Arithmetic
- Pointer and Array Relationship
- Pointers to Functions
- Pointers to Pointers (int **ptr)
- Pointers with Strings and Structures

9. Structures and Unions

- Declaring and Accessing Structures
- Structure Initialization
- Nested Structures
- Array of Structures
- Passing Structure to Functions
- typedef, enum
- Unions – Memory Sharing
- Structure vs Union

10. Dynamic Memory Allocation

- Introduction to DMA
- malloc(), calloc(), realloc(), free()
- Difference Between Static and Dynamic Allocation
- Memory Leak and Best Practices

11. File Handling

- File Pointer and Modes: r, w, a, rb, wb, etc.
- fopen(), fclose(), fscanf(), fprintf()
- fgetc(), fputc(), fgets(), fputs()
- Reading/Writing Characters, Lines, and Records
- File Error Handling

PART B: C++ Programming

1. Introduction to C++

- History and Features of C++
- Difference Between C and C++
- Procedural vs Object-Oriented Paradigm
- Structure of a C++ Program
- Input/Output in C++: cin, cout

2. Basics of Object-Oriented Programming (OOP)

- What is OOP?
- **4 Pillars of OOP:**
    - Encapsulation
    - Abstraction
    - Inheritance
    - Polymorphism
- Classes and Objects – Definition, Accessing Members
- Access Specifiers: private, public, protected

3. Constructors and Destructors

- Constructor: Default, Parameterized, Copy Constructor
- Constructor Overloading
- Destructor: Purpose and Use Case

4. Functions in C++

- Member Functions
- Function Overloading
- Inline Functions
- Static Members and Static Functions
- Friend Functions (Accessing Private Members)

5. Inheritance

- **Types:**
    - Single
    - Multilevel
    - Multiple

    - Hierarchical
    - Hybrid
- Constructor Calling Order
- Overriding and Base-Class Access
- Virtual Base Class (Diamond Problem)

6. Polymorphism

- **Compile-Time:**
    - Function Overloading
    - Operator Overloading
- **Run-Time:**
    - Virtual Functions
    - Pure Virtual Functions
    - Abstract Class
- this Pointer

7. Pointers in C++

- Basics and Object Pointers
- this Pointer
- Pointers to Derived Class
- Dynamic Object Creation using new and delete

8. File Handling in C++

- ifstream, ofstream, fstream
- Reading/Writing to File
- File Open Modes

9. Exception Handling

- try, catch, throw
- Multiple catch Blocks
- Catch All Exception (catch(...))
- Re-throwing Exception

10. Introduction to STL

- Need for STL
- **Components:**
    - Containers (Vector, List, Stack, Queue, Map, Set, Heap (Priority queue))
    - Iterators
    - Algorithms (sort(), find(), etc.)
- Hands-on with Vector and Map

Note :

- Multiple Questions Practice after every unit
- Short Coding Assignments