TECHNOCATION FREELANCING TRAINING INSTITUTE & SOFTWARE HOUSE

Professional C Programming Course Outline

Module 1: Introduction to C Programming

- Overview of C Programming & History
- Setting Up C Development Environment (GCC, Code::Blocks, Visual Studio)
- Writing & Compiling a Simple C Program (`hello.c`)
- Understanding the Compilation Process (Preprocessing, Compilation, Linking)
- Structure of a C Program (`main()`, Libraries, `#include`, Comments)

Module 2: C Fundamentals & Data Types

- C Syntax & Basic Program Structure
- Variables & Constants (`int`, `float`, `char`, `double`)
- Operators in C (Arithmetic, Logical, Bitwise, Assignment)
- Input & Output Functions (`printf()`, `scanf()`, `gets()`, `puts()`)
- Type Conversion & Typecasting

Module 3: Control Flow & Decision Making

- Conditional Statements (`if`, `else if`, `switch-case`)
- Looping Structures (`for`, `while`, `do-while`)
- `break`, `continue`, & `goto` Statements
- Nested Loops & Conditional Logic
- Using `enum` for Readable Code

Module 4: Functions & Modular Programming

- Defining & Calling Functions
- Function Parameters & Return Values
- Recursion & Recursive Functions
- Scope & Lifetime of Variables (Global vs. Local)
- Storage Classes (`auto`, `static`, `register`, `extern`)

Module 5: Arrays & Strings in C

- Declaring & Initializing Arrays (1D & Multi-Dimensional)
- Accessing & Manipulating Array Elements
- Introduction to Strings & Character Arrays
- String Handling Functions (`strlen()`, `strcpy()`, `strcmp()`, `strcat()`)
- Pointers & Arrays (Pointer Arithmetic)

Module 6: Pointers & Memory Management

- Understanding Pointers & Memory Addresses
- Pointer Arithmetic (`*`, `&`, `->`)
- Dynamic Memory Allocation (`malloc()`, `calloc()`, `realloc()`, `free()`)
- Dangling Pointers & Memory Leaks
- Function Pointers & Callbacks

Module 7: Structures & Unions in C

- Defining & Using `struct` (Structures)
- Nested Structures & Arrays of Structures
- `typedef` & `enum` for Custom Data Types
- Understanding `union` & Its Memory Efficiency
- Passing Structures to Functions

Module 8: File Handling & I/O Operations

- File Handling Basics (`fopen()`, `fclose()`, `fprintf()`, `fscanf()`)
- Reading & Writing Files (Text & Binary)
- File Pointers & Seeking (`fseek()`, `ftell()`, `rewind()`)
- Error Handling in File Operations

- Working with Command Line Arguments

Module 9: Advanced C Programming Concepts

- Bitwise Operations & Manipulation
- Preprocessors & Macros (`#define`, `#ifdef`, `#ifndef`)
- Multi-Threading & Concurrency in C
- Implementing Data Structures (Linked List, Stack, Queue)
- Signal Handling & Exception Handling in C

Module 10: Real-World Projects & Optimization

- Developing a Simple Banking System
- Building a Student Management System
- Creating a Basic File Compression Tool
- Optimizing Code for Performance & Memory Efficiency
- Debugging Techniques & Tools (`gdb`, `valgrind`)