C - Training syllabus

Fundamentals in C

- Program
- Programming
- Programming Languages
- Types of software
- Introduction to C
- History of C
- Features of C
- Applications of C
- Character set, ASCII Table
- Tokens
- Keywords
- Identifiers & Naming Rules
- constants
- Data Types
- Type Qualifiers
- How does the data stored in Computers Memory
- Variables
- Variable Declaration
- Variable Assignment
- Variable Initialization
- Comments
- Defining Constants
- MCQs

Operators and Expressions

- Arithmetic operators
- Arithmetic expressions
- Evaluation of expressions
- Relational operators
- Logical operators
- Assignment operators
- Increment & decrement operators
- Conditional operator
- Bitwise operators

- Type casting
- Sizeof operator
- Comma operator
- Operators Precedence and Associativity
- Expressions
- Evaluation of Expressions
- MCQs

Input-Output Functions

- Input-Output Library Functions
- Non-formatted Input and Output
- Character oriented Library functions
- Compiler, Linker and Loader
- Program execution phases
- Formatted Library Functions
- Mathematical Library Functions
- Structure of a C Program
- IDE
- Basic programs
- MCQs

Control Statements

- Conditional Control Statements
- If
- if-else
- nested if-else
- if-else-if ladder
- Multiple Branching Control Structure
- switch-case
- Loop Control statements
- While
- do-while
- for
- Nested Loops
- Jump Control structures
- break
- continue
- goto
- return
- Programs
- MCQs

Arrays

- Arrays
- One dimensional arrays
- Declaration of 1D arrays
- Initialization of 1D arrays
- Accessing element of 1D arrays
- Reading and displaying elements
- Programs on 1D Arrays
- Two dimensional arrays
- Declaration of 2D arrays
- Initialization of 2D arrays
- Accessing element of 2D arrays
- Reading and displaying elements
- Programs on 2D Arrays
- Three dimensional arrays
- MCQs

Strings

- String Concept
- Introduction to String in C
- Storing Strings
- The string Delimiter
- String Literals (String Constants)
- Strings and Characters
- Declaring Strings
- Initializing Strings
- Strings and the Assignment Operator
- String Input Functions / Reading Strings
- String Output Functions / Writing Strings
- String Input-Output using fscanf() and fprintf() Functions
- Single Character Library Functions / Character Manipulation in the String
- String Manipulation Library Functions
- Programs Using Character Arrays
- Array of Strings (2D Character Arrays)
- Programs Using Array of Strings
- MCQs

Pointers

- Understanding Memory Addresses
- Pointer Operators
- Pointer

- Pointer Advantages and Disadvantages
- Declaration of Pointer Variables
- Initialization of Pointer Variables
- Dereferencing / Redirecting Pointer Variables
- Declaration versus Redirection
- Void Pointer
- Null Pointer
- Compatibility
- Array of Pointers
- Pointer to Pointer
- Pointer Arithmetic
- Dynamic Memory Allocation Functions

Functions

- Functions
- Advantages of using functions
- Defining a function
- Calling a function
- Return statement
- Function Prototype
- Basic Function Designs
- Programs Using Functions
- Scope
- Recursion
- Iteration vs Recursion
- Nested functions
- Variable Length Number of Arguments
- Parameter Passing Techniques – Call by value & Call by Address
- Functions Returning Pointers
- Pointers and One-Dimensional Arrays
- Pointers and Two-Dimensional Arrays
- Passing 1D arrays to Functions
- Passing 2D arrays to Functions
- Pointers and Strings
- Passing Strings to Functions
- Pointer to Function
- MCQs

Storage Classes

- Object Attributes
- Scope
- Extent

- Linkage
- auto
- static
- extern
- register
- MCQs

Preprocessor Directives

- The #include Preprocessor Directive & User defined header files
- The #define Preprocessor Directive: Symbolic Constants
- The #define Preprocessor Directive: Macros
- Conditional Compilation Directives
- #if
- #else
- #elif
- #endif
- #ifdef
- #ifndef
- #undef
- #error
- #line
- #pragma
- MCQs

Structures, Unions, Enumerations and Typedef

- Structures
- Structure Type Declaration
- Structure Variable Declaration
- Initialization of Structure
- Accessing the members of a structure
- Programs Using Structures
- Operations on Structures (Copying and Comparing Structures)
- Nested structures (Complex Structures)
- Structures Containing Arrays (Complex Structures)
- Array of Structures (Complex Structures)
- Pointer to Structure
- Accessing structure member through pointer using dynamic memory allocation
- Pointers within Structures
- Self-referential structures
- Passing Structures to Functions
- Functions returning Structures
- Unions
- Differences between Structures & Unions

- Enumerated Types / enum keyword
- The Type Definition / typedef keyword
- Bit fields
- MCQs

Command Line Arguments

Files

- Concept of a file
- Streams
- Text File and Binary Files
- State of a File
- Opening and Closing Files
- File Input / Output Functions
- Formatted Input-Output Functions
- Character Input-Output Functions
- Line Input-Output Functions
- Block Input-Output Functions
- File Status Functions (Error Handling)
- Positioning Functions
- System File Operations
- MCQs

Graphics

- Initialization of graphics
- Drawing shapes using pre-defined functions
- Finding the resolution of screen
- Setting colors to text and window
- Font settings
- Fill styles
- Basic GUI applications