FAIZ COMPUTER INSTITUTE

C Programming Syllabus

1. Introduction to C Programming

- Overview of C programming language
- History of C language
- Importance and applications of C
- Setting up the development environment (IDE, Compiler)
- Writing and running your first C program

2. Basic Syntax and Structure

- Structure of a C program (main function, header files)
- Keywords, Identifiers, and Data Types
- Constants and variables
- Operators: Arithmetic, Relational, Logical, Bitwise, Assignment
- Input and Output: scanf(), printf()

3. Control Structures

- Conditional Statements: if, else, switch
- Looping Statements: for, while, do-while
- Nested loops and conditional statements

4. Functions

- Defining functions
- Function declaration, definition, and calling
- Function arguments (passing by value and reference)
- Return types
- Recursive functions
- Scope and lifetime of variables (local vs global)

5. Arrays

- One-dimensional arrays
- Two-dimensional arrays
- Array initialization and manipulation
- Multidimensional arrays
- Passing arrays to functions

6. Pointers

- Introduction to pointers
- Pointer operations (dereferencing, referencing)
- Pointers and arrays

- Pointers to functions
- Dynamic memory allocation (malloc(), calloc(), free())

7. Strings

- String handling in C
- String manipulation functions (strcpy(), strlen(), strcmp(), etc.)
- Character arrays vs string literals
- Passing strings to functions

8. Structures and Unions

- Defining and using structures
- Accessing structure members
- Arrays of structures
- Nested structures
- Introduction to unions
- Differences between structures and unions

9. File Handling

- Introduction to file handling in C
- File operations: Opening, closing, reading, writing files
- File pointers (FILE *)
- Reading and writing text files
- Binary file handling

10. Dynamic Memory Allocation

- Memory management functions: malloc(), calloc(), realloc(), free()
- Memory leaks and avoiding them
- Fragmentation

11. Error Handling

- Understanding errors and debugging
- Using errno, perror(), and strerror()
- Writing error messages to files
- Handling exceptions in C

15. Final Projects

- Project Ideas:
    - Develop a simple calculator program
    - Create a student record management system (file-based)
    - Build a library management system using structures
    - Implement a basic banking system with file handling
    - Write a program to implement sorting algorithms