

MODUL2:- Introduction to Programming

LAB EXERCISE:

Q1:- Research and provide three real-world applications where C programming is extensively used, such as in embedded systems, operating systems, or game development.

- ⇒ C programming is extensively used in various domains, including operating systems, where it forms the backbone of systems like Unix and Linux. Additionally, it plays a crucial role in embedded systems, powering devices such as microcontrollers and IoT devices, and is widely utilized in game development for performance-intensive applications.

1. Operating Systems

- ⇒ C programming is fundamental in the development of operating systems.
- ⇒ Notable examples include:
- ⇒ **Unix**: The first operating system written in C, which set the standard for many others.
- ⇒ **Linux**: A widely used open-source operating system that relies heavily on C for its kernel and system utilities.
- ⇒ **Windows**: Portions of Microsoft Windows are also developed using C, showcasing its versatility in system-level programming.

2. Embedded Systems

- ⇒ C is the preferred language for programming embedded systems due to its efficiency and close-to-hardware capabilities.
- ⇒ Applications include:
- ⇒ **Microcontrollers**: Used in various consumer electronics, automotive systems, and industrial machines.
- ⇒ **IoT Devices**: Many Internet of Things devices utilize C for their firmware, enabling them to operate efficiently with limited resources.
- ⇒ **Automotive Systems**: C is used in the development of software for engine control units (ECUs) and other critical automotive applications.

3. Game Development

- ⇒ C programming is extensively used in the gaming industry for its performance and control over system resources.
- ⇒ Key aspects include:
- ⇒ **Game Engines**: Many popular game engines, such as Unreal Engine, are developed using C or C++ for high-performance graphics and real-time processing.
- ⇒ **Game Logic**: Core game mechanics and systems are often implemented in C to ensure optimal performance.

⇒ **Cross-Platform Development:** C allows for easier porting of games across different platforms, including consoles and PCs.

Q2:- Install a C compiler on your system and configure the IDE. Write your first program to print "Hello, World!" and run it.

[lab_prectic\hello.c](#)

q3:- Write a C program that includes variables, constants, and comments. Declare and use different data types (int, char, float) and display their values.

[lab_prectic\variable.c](#)

[lab_prectic\operater.exe](#)

q4:- Write a C program that accepts two integers from the user and performs arithmetic, relational, and logical operations on them. Display the results.

[lab_prectic\operater.c](#)

[lab_prectic\operater.exe](#)

q5:- Write a C program to check if a number is even or odd using an if-else statement. Extend the program using a switch statement to display the month name based on the user's input (1 for January, 2 for February, etc.).

[lab_prectic\q5.c](#)

Q6:- Write a C program to print numbers from 1 to 10 using all three types of loops (while, for, do-while).

[lab_prectic\for_loop.c](#)

Q7:- Write a C program that uses the break statement to stop printing numbers when it reaches 5. Modify the program to skip printing the number 3 using the continue statement.

[lab_prectic\q7.c](#)

Q8:- Write a C program that calculates the factorial of a number using a function. Include function declaration, definition, and call.

[lab_prectic\fact_num.c](#)

Q9:- Write a C program that stores 5 integers in a one-dimensional array and prints them. Extend this to handle a two-dimensional array (3x3 matrix) and calculate the sum of all elements.

[lab_prectic\print_array.c](#)

Q10:- Write a C program to demonstrate pointer usage. Use a pointer to modify the value of a variable and print the result.

[lab_prectic\pointer.c](#)

Q11:- Write a C program that takes two strings from the user and concatenates them using strcat(). Display the concatenated string and its length using strlen().

[lab_prectic\concat leght string.c](#)

Q12:- Write a C program that defines a structure to store a student's details (name, roll number, and marks). Use an array of structures to store details of 3 students and print them.

[lab_prectic\struct detail.c](#)

Q13:- Write a C program to create a file, write a string into it, close the file, then open the file again to read and display its contents.

[lab_prectic\fill hendaling.c](#)