

SAVEETHA SCHOOL OF ENGINEERING SAVEETHA INSTITUTE OF MEDICAL AND TECHNICAL SCIENCES

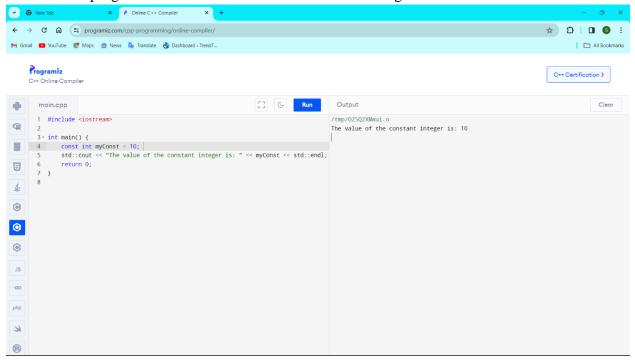


SUB CODE & NAME: DSA01/Object Oriented Programming with C++

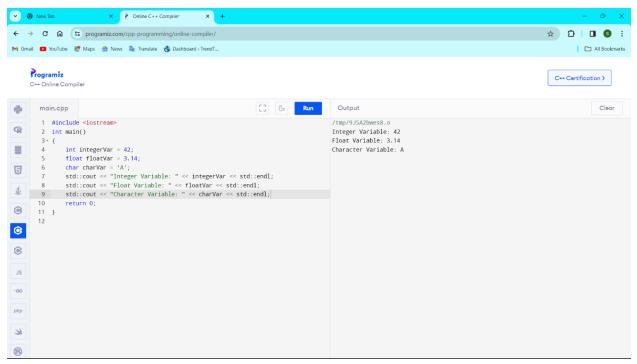
LAB DAY 1/13-03-2024

EASY

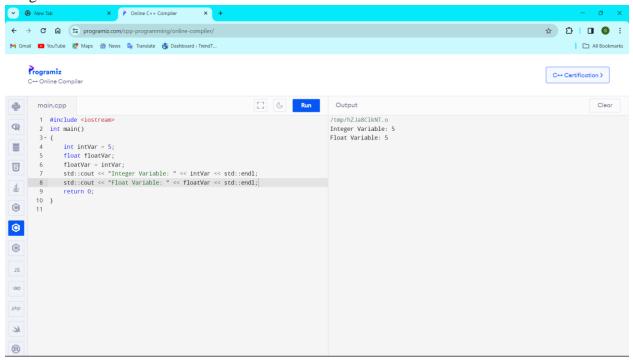
1. Write a C++ program to declare and initialize a constant integer with value 10.



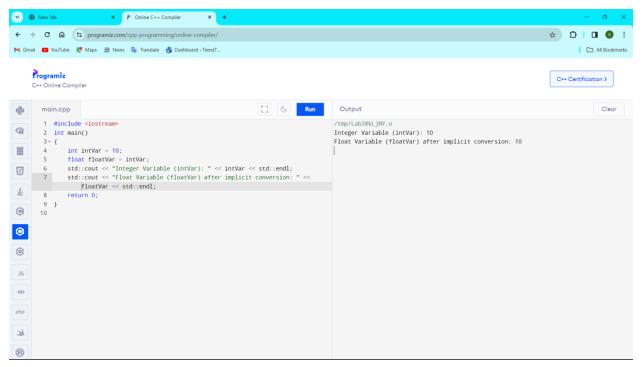
2. Write a C++ program to demonstrate the use of integer, float, and character data types



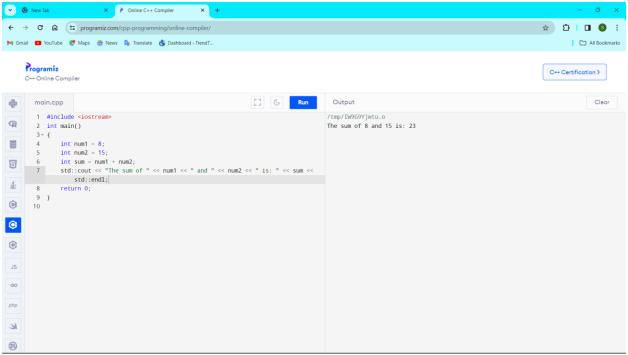
3. Create a C++ program to declare an integer variable and a float variable, then assign the integer variable to the float variable



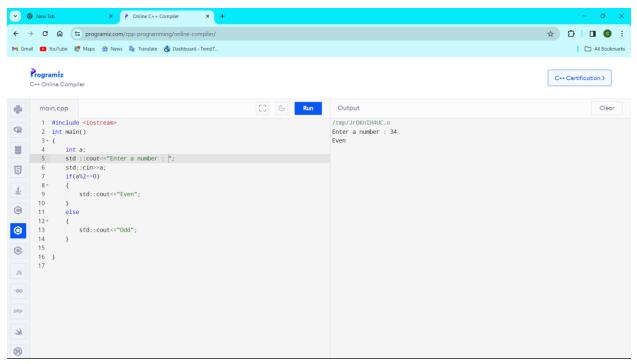
4. Implement a C++ program that performs an implicit conversion from integer to float



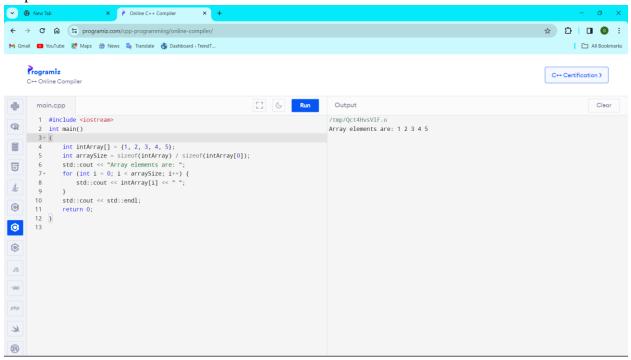
5. Write a C++ program to declare and initialize two integer variables and perform addition using the + operator



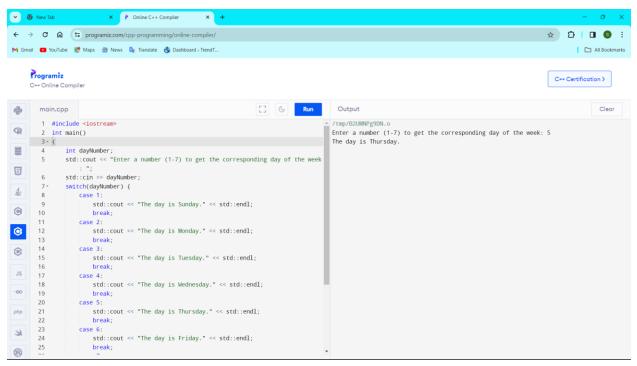
6. Create a C++ program to demonstrate the use of if-else control structure to determine if a number is even or odd



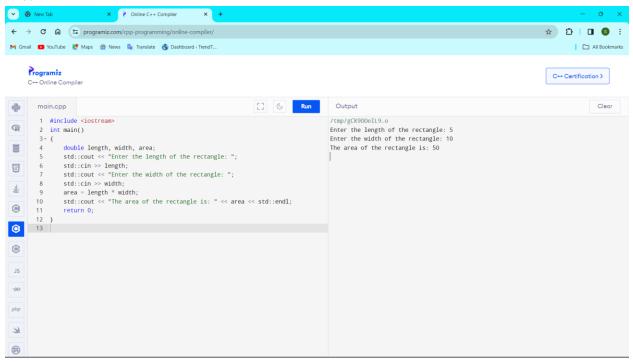
7. Write a C++ program to declare an array of integers and display its elements using a loop.



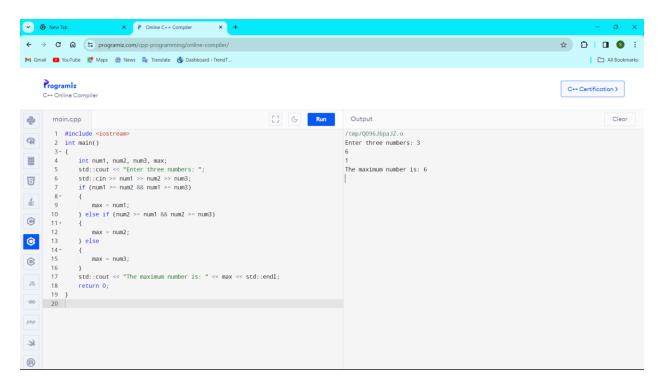
8. Implement a C++ program to demonstrate the use of switch-case control structure to display the name of the day based on the user input (1-7).



9. Create a C++ program to calculate the area of a rectangle using user input for length and width

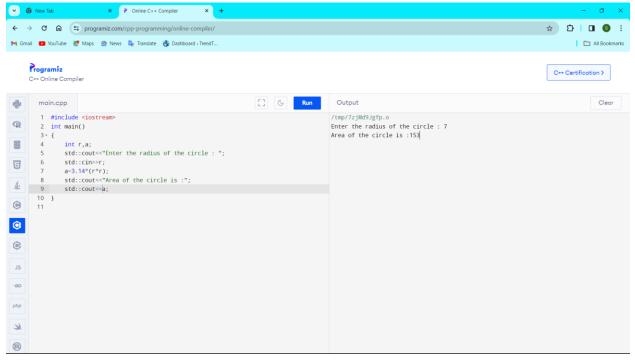


10. Write a C++ program to find the maximum of three numbers using conditional statements.

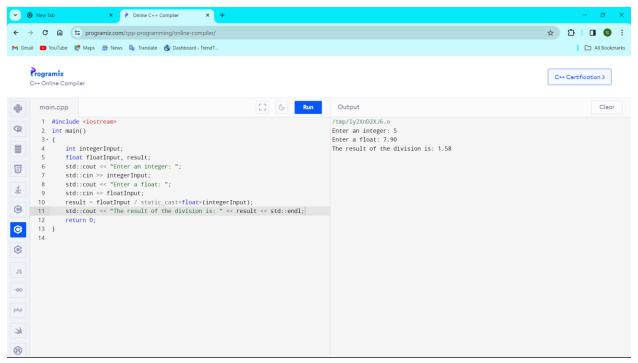


MEDIUM

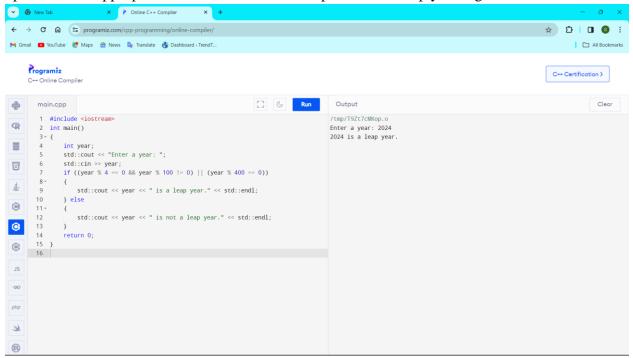
1. Write a C++ program that calculates the area of a circle. Use a constant variable for the value of pi and prompt the user to enter the radius. Display the calculated area.



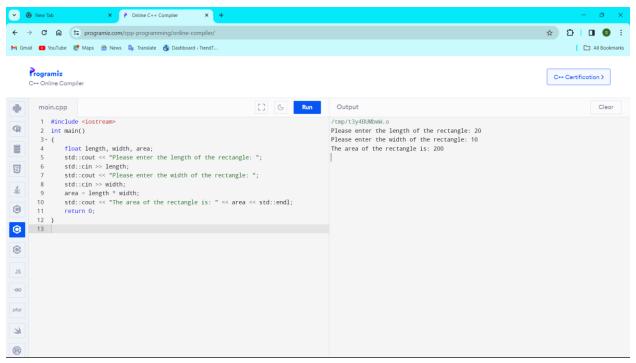
2. Develop a C++ program that reads two numbers from the user, one integer and one float. Perform division of the float by the integer and display the result. Ensure proper type compatibility and implicit conversion handling.



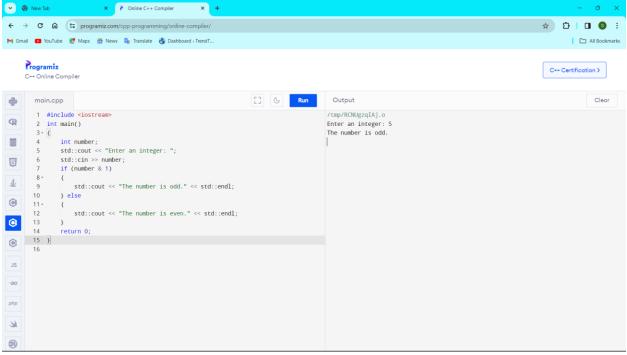
3. Write a C++ program to determine whether a given year is a leap year or not. Use logical operators and appropriate control structures to implement the leap year logic.



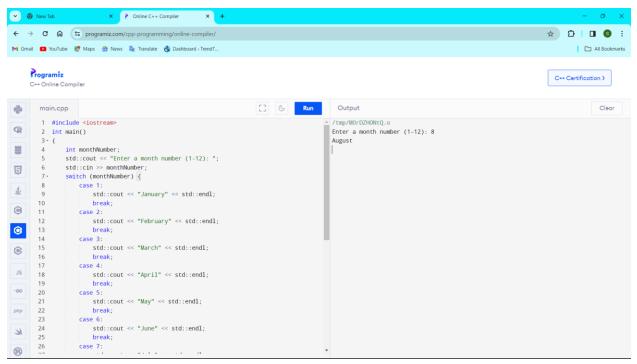
4. Create a C++ program that calculates the area of a rectangle. Prompt the user to enter the length and width of the rectangle as floating-point numbers, and display the calculated area.



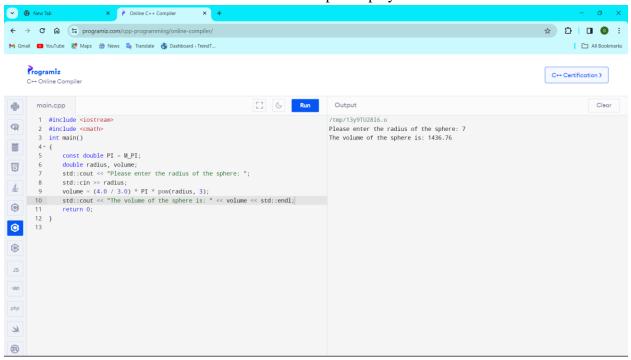
5. Develop a C++ program that reads an integer from the user and checks if it is an odd number. Use bitwise AND operator and handle implicit conversion properly.



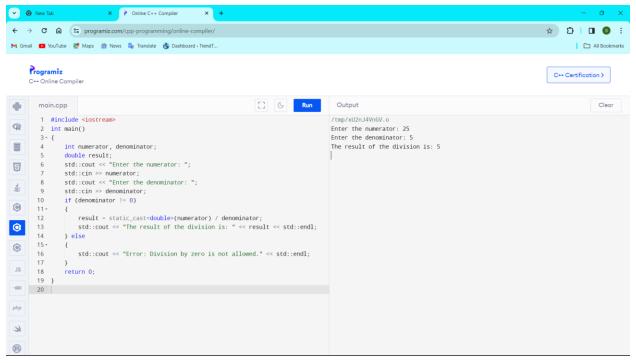
6. Write a C++ program that prompts the user to enter a month number (1-12) and displays the corresponding month name. Use a switch-case statement to implement this.



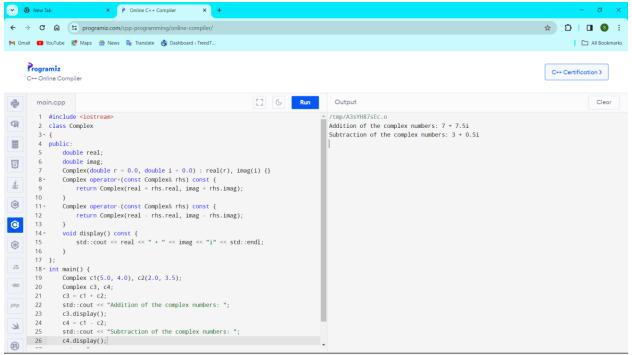
7. Create a C++ program that calculates the volume of a sphere. Prompt the user to enter the radius and use a constant variable for the value of pi. Display the calculated volume.



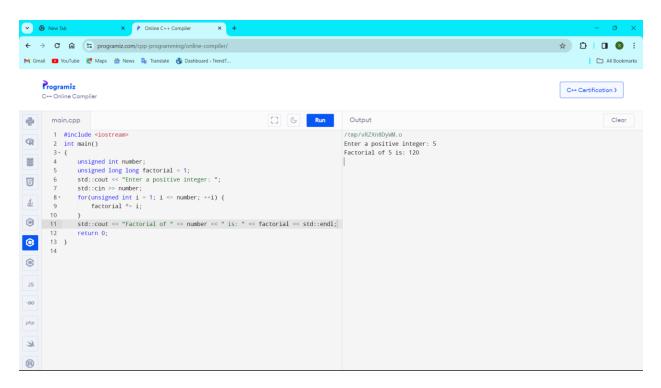
8. Develop a C++ program that reads two integers from the user and performs division. Ensure that if the second number is zero, the program should not crash, but display an appropriate error message.



9. Write a C++ program that defines a class representing a complex number. Implement operator overloading for addition and subtraction of complex numbers.

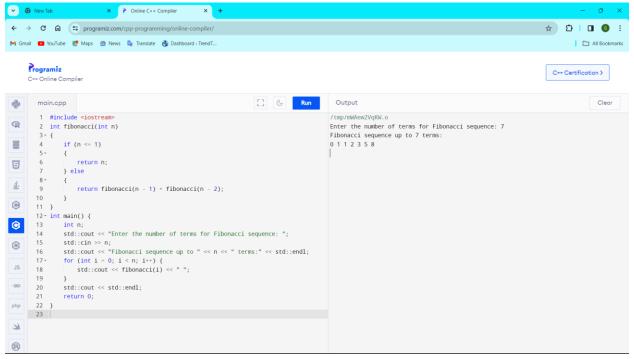


10. Create a C++ program that prompts the user to enter a number and calculates the factorial of that number using a loop. Display the result.

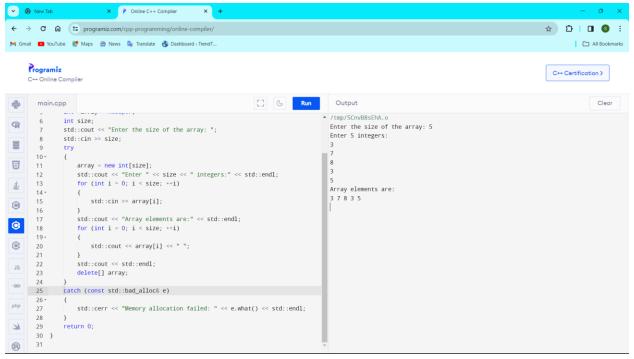


HARD

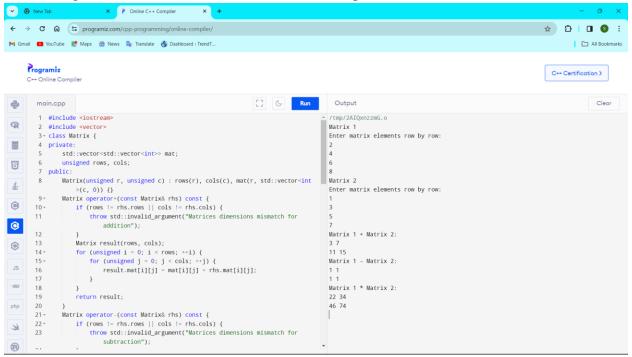
1. Write a C++ program to implement a recursive function to calculate the Fibonacci sequence up to a given number 'n'.



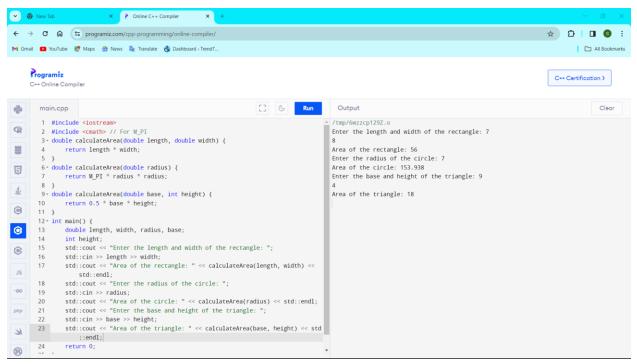
2. Develop a C++ program that dynamically allocates memory for an array of integers based on user input for the array size. Implement error handling for memory allocation failure.



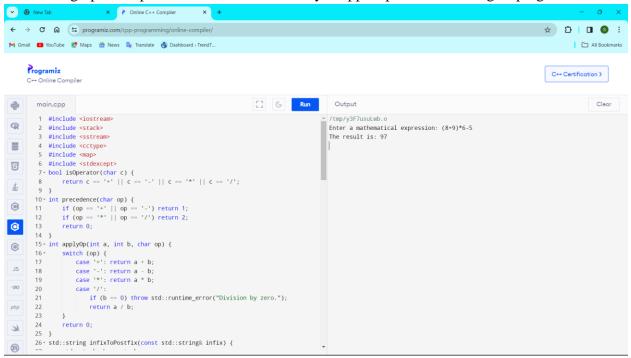
3. Create a C++ program that defines a class representing a matrix. Implement operator overloading for matrix addition, subtraction, and multiplication.



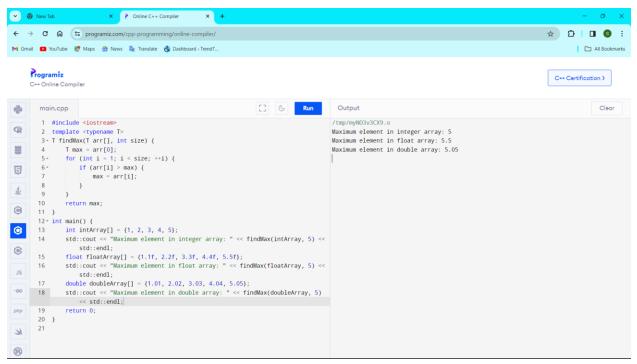
4. Write a C++ program that demonstrates function overloading for different data types. Create overloaded functions to calculate the area of a rectangle, circle, and triangle.



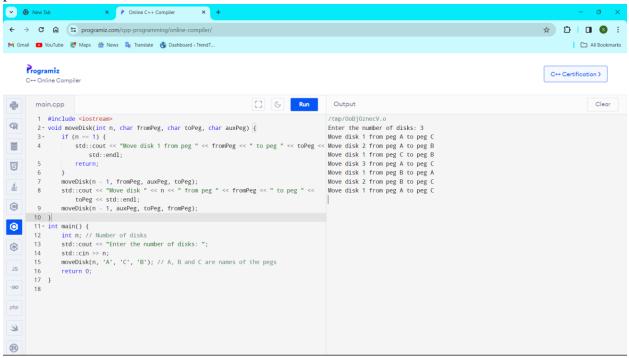
5. Develop a C++ program that evaluates mathematical expressions entered by the user, considering operator precedence and associativity. Support parentheses for grouping.



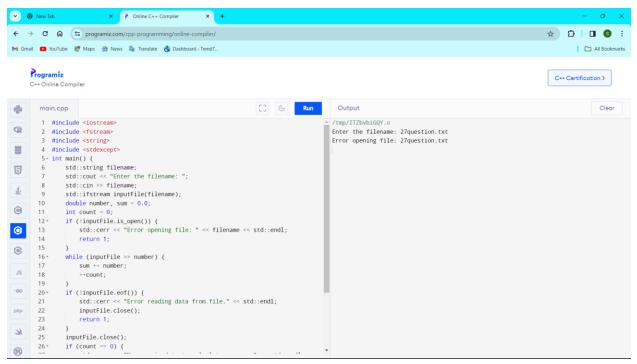
6. Write a C++ program that defines a template function to find the maximum element in an array of any data type. Test the function with integer, float, and double arrays.



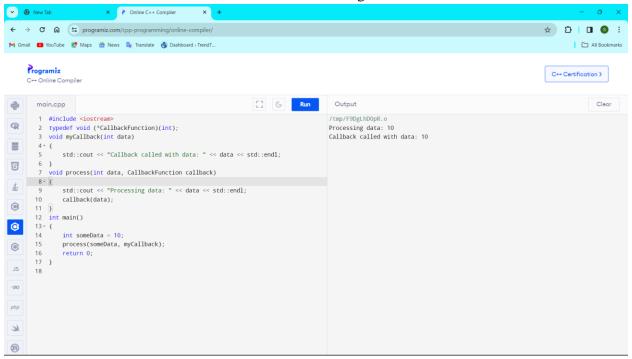
7. Create a C++ program to implement a recursive function to solve the Tower of Hanoi puzzle with 'n' disks.



8. Develop a C++ program that reads input from a file containing numeric data and calculates the average. Implement exception handling to handle file I/O errors and invalid data.



9. Write a C++ program that implements a callback mechanism using function pointers. Create a function that takes a callback function as an argument and calls it.



10. Develop a C++ program to implement a binary search tree (BST) and perform operations such as insertion, deletion, and traversal (in-order, pre-order, post-order).

