INSTITUTE OF TECHNOLOGY AND MANAGEMENT
SKILLS UNIVERSITY,
KHARGHAR, NAVI MUMBAI

C++ PROGRAMMING LAB

Prepared by:

Name of Student: _Shikha singh_____

Roll No: 25_____

Batch: 2023-27

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Exp. No	List of Experiment
	Write a program to find the roots of a quadratic equation.
	Write a program to calculate the power of a number using a loop.
	Write a program to check if a given string, is a palindrome.
	Write a program that simulates a simple ATM machine, allowing users to check their balance, deposit, or withdraw money using a switch statement.
	Write a program that finds the largest among three numbers using nested if-else statements
	Write a program that determines the grade of a student based on their marks of 5 subjects using if-else-if ladder.
	Write a program to find the sum of digits of a number until it becomes a single-digit number.
	Write a program to print a Pascal's triangle using nested loops.
	Write a program to calculate the sum of series 1/1! + 2/2! + 3/3! + ... + N/N! using nested loops.
0	Write a program to create an array of strings and display them in alphabetical order.
1	Write a program that checks if an array is sorted in ascending order.
2	Write a program to calculate the sum of elements in each row of a matrix.
3	Write a program to generate all possible permutations of a string.
4	Create a C++ program to print the following pattern: ***** * * * * * * *****
5	Write a C++ program to display the following pattern: 1 232

	34543 4567654 34543 232

6	Write a program to creating an inventory management system for a small store. The system should use object-oriented principles in C++. Your program should have the following features:
	• Create a **Product** class that represents a product in the inventory. Each **Product** object should have the following attributes:
	  • Product ID (an integer)
	  • Product Name (a string)
	  • Price (a floating-point number)
	  • Quantity in stock (an integer)
	• Implement a parameterized constructor for the **Product** class to initialize the attributes when a new product is added to the inventory.

7	Write a program to manage student records. Create a class Student with attributes such as name, roll number, and marks. Implement methods for displaying student details, adding new students, and calculating the average marks of all students in the record system.
8	Write a program that implements a basic calculator. Use a class Calculator with methods to perform addition, subtraction, multiplication, and division of two numbers. The program should allow the user to input two numbers and select an operation to perform.
9	Write a program to simulate a simple online shop. Create a class Product with attributes like name, price, and quantity in stock. Implement methods for adding products to the shopping cart, calculating the total cost, and displaying the contents of the cart.
0	Write a program to manage student grades for a classroom. Create a class Student with attributes for student name and an array to store grades. Implement methods for adding grades, calculating the average grade, and displaying the student's name and grades. Use constructors and destructors to initialize and release resources.

Name of Student: Shikha singh_____

Roll Number: _25_____

Experiment No: 18.

Title: 18. Write a program that implements a basic calculator. Use a class Calculator with methods to perform addition, subtraction, multiplication, and division of two numbers. The program should allow the user to input two numbers and select an operation to perform.

Theory:

Calculator Class:

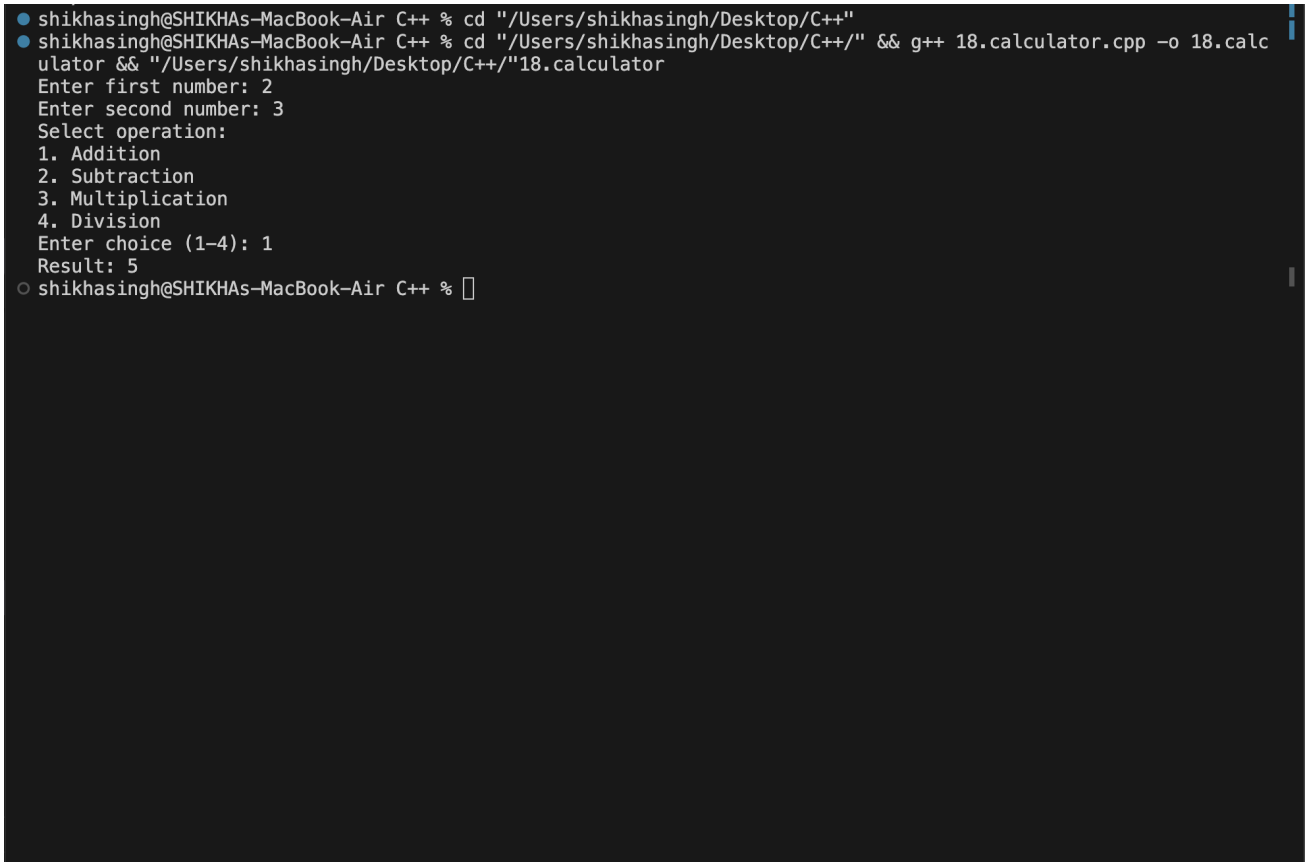
- **The program defines a Calculator class that encapsulates the basic arithmetic operations - addition, subtraction, multiplication, and division.**
- **Each operation is implemented as a member function of the class.**
- **Main Function:**
- **In the main function, an instance of the Calculator class is created.**
- **User Input:**
- **The program prompts the user to input two numbers and stores them in variables num1 and num2.**
- **The user is also asked to select an operation by entering a choice (1 for addition, 2 for subtraction, and so on).**
- **Operation Selection:**
- **A switch statement is used to determine which operation to perform based on the user's choice.**
- **The selected operation is then performed using the corresponding method of the Calculator class.**
- **Error Handling:**
- **The program checks for division by zero and handles it by displaying an error message and returning 0 in such cases.**

Code:

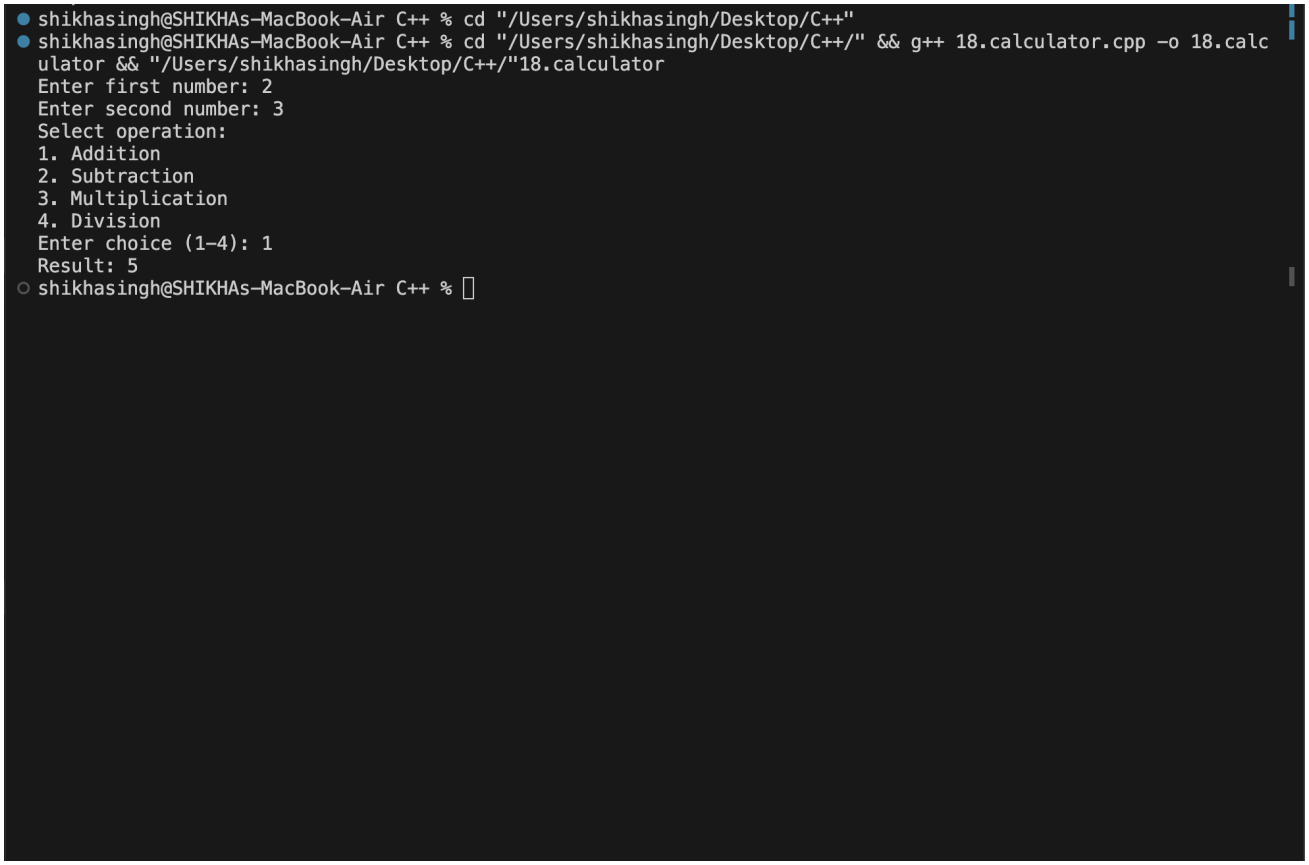
```
#include <iostream>

using namespace std;
class Calculator {
public:
```

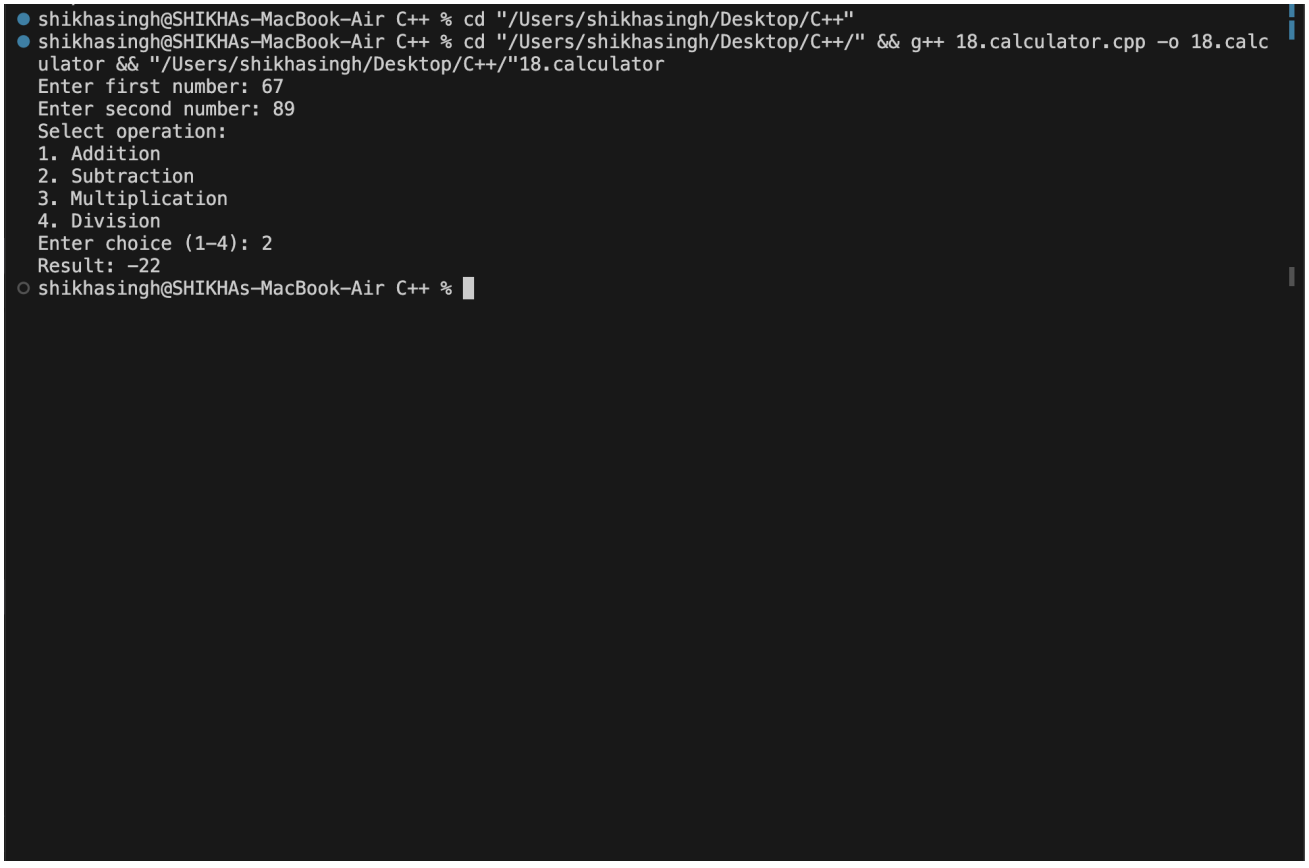
```
    double add(double num1, double num2) {
        return num1 + num2;
    }
    double subtract(double num1, double num2) {
        return num1 - num2;
    }
    double multiply(double num1, double num2) {
        return num1 * num2;
    }
    double divide(double num1, double num2) {
        if (num2 != 0) {
            return num1 / num2;
        } else {
            cout << "Error: Division by zero is not allowed." <<endl;
            return 0.0;
        }
    }
};
int main() {

    Calculator calculator;
    double num1, num2;
   cout << "Enter first number: ";
    cin >> num1;
    cout << "Enter second number: ";
    cin >> num2;
    int choice;
    cout << "Select operation:\n1. Addition\n2. Subtraction\n3. Multiplication\n4. Division\nEnter choice (1-4): ";
    cin >> choice;
    switch (choice) {
        case 1:
            cout << "Result: " << calculator.add(num1, num2) << endl;
            break;
        case 2:
            cout << "Result: " << calculator.subtract(num1, num2) <<endl;
            break;
        case 3:
            cout << "Result: " << calculator.multiply(num1, num2) <<endl;
            break;
        case 4:
            cout << "Result: " << calculator.divide(num1, num2) <<endl;
            break;
        default:
            cout << "Invalid choice. Please enter a number between 1 and 4." << endl;
            break;
    }
    return 0;
}
```

Output: (screenshot)



Test Case: Any two (screenshot)





Conclusion:

This basic C++ program demonstrates the principles of object-oriented programming by using a class (Calculator) to organize related functionality. It allows the user to perform basic arithmetic operations on two numbers. The use of a switch statement facilitates the selection of the desired operation.

The program also includes a simple form of error handling to address the possibility of division by zero. However, for more robust error handling, additional measures and validation could be implemented.