Meeting 7

Avicena Yosanta Muchtar 24/536710/PA/22764

22 October 2024

1 Question 1:

Question:
Write a program with some function to calculate simple mathematic operation, namely addition or difference!

Answer:

```cpp

#include <iostream>
using namespace std;

int add(int a, int b) {
    return a + b;
}

int subtract(int a, int b) {
    return a - b;
}

int main() {
    int num1, num2, choice;

    cout << "Enter two numbers: ";
    cin >> num1 >> num2;

    cout << "Choose the operation:\n1. Addition\n2.
    ↪  Subtraction\n";
    cin >> choice;

    if (choice == 1) {
        cout << "Result: " << add(num1, num2) << endl;
    } else if (choice == 2) {
```

```
        cout << "Result: " << subtract(num1, num2) << endl;
    } else {
        cout << "Invalid choice" << endl;
    }

    return 0;
}
```

2 Explanation

```
int add(int a, int b) {
    return a + b;
}

int subtract(int a, int b) {
    return a - b;
}

```

- Add(int a, int b): A function that takes two integer number (a and b) as input and returns the sum of the number.
- subtract(int a, int b): A function that takes two integer number as input and returns their difference (a - b).

```
int main() {
    int num1, num2, choice;
```

Three integer variables are declared:
- num1: First number input entered by the user.
- num2: Second number input entered by the user.
- choice: Used to select the choice for addition or subtraction.

```
    cout << "Enter two numbers: ";
    cin >> num1 >> num2;
```

The program ask the user to enter two numbers.
cin >> num1 >> num2; store the numbers and stores them in the variables num1 and num2.

```
    if (choice == 1) {
        cout << "Result: " << add(num1, num2) << endl;
    } else if (choice == 2) {
```

```
4           cout << "Result: " << subtract(num1, num2) << endl;
5       } else {
6           cout << "Invalid choice" << endl;
7       }
```

The if statement checks the value of choice:
- If choice is 1, the program use the add() function with num1 and num2 as arguments and prints the result.
- If choice is 2, the program use the subtract() function and prints the result.
- If the user enters anything other than 1 or 2, the program will print \Invalid choice."

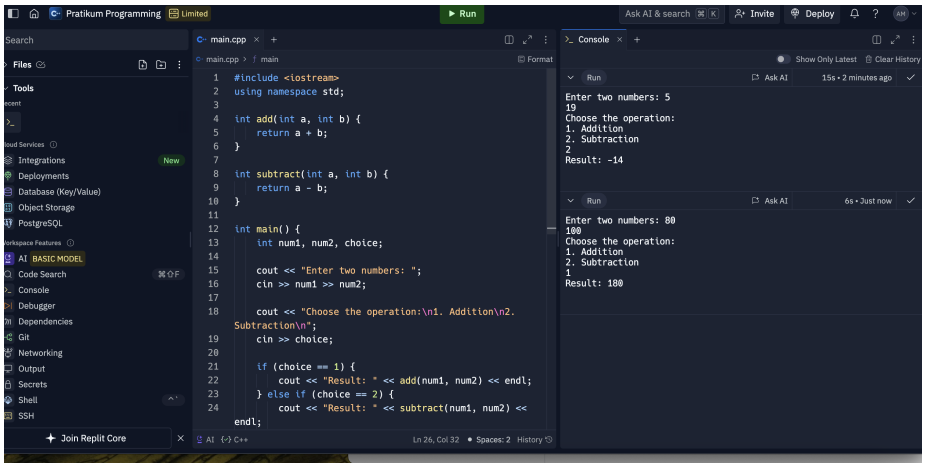

Figure 1: Ouput

3 Questio 2:

Question:
Write a program with recurrence function to calculate x! (x factorial)!

```cpp
#include <iostream>
using namespace std;

int factorial(int x) {
    if (x == 0 || x == 1) {
        return 1;
    } else {
        return x * factorial(x - 1);
    }
}

int main() {
    int num;

    cout << "Enter a number to calculate its factorial: ";
    cin >> num;

    if (num < 0) {
        cout << "Factorial is not defined for negative numbers."
        ↪  << endl;
    } else {
        cout << "Factorial of " << num << " is: " <<
        ↪  factorial(num) << endl;
```

```
    }

    return 0;
}
```

4 Explanation

```
    int factorial(int x) {
    if (x == 0 || x == 1) {
        return 1;
    } else {
        return x * factorial(x - 1);
    }
}
```

- When x == 0 or x == 1, the function returns 1. This is the stopping point for the recursion.
- If x is greater than 1, the function use (factorial(x - 1)) and multiplies x by the factorial of x - 1. This will go on until the base case of the program is reached.

```
    int main() {
    int num;

    cout << "Enter a number to calculate its factorial: ";
    cin >> num;

    if (num < 0) {
        cout << "Factorial is not defined for negative numbers."
        ↪  << endl;
    } else {
        cout << "Factorial of " << num << " is: " <<
        ↪  factorial(num) << endl;
    }

    return 0;
}
```

User input:
- cout << "Enter a number to calculate its factorial: ";: Displays text asking the user to enter a number.
- cin >> num;: The user input is stored in the num variable.

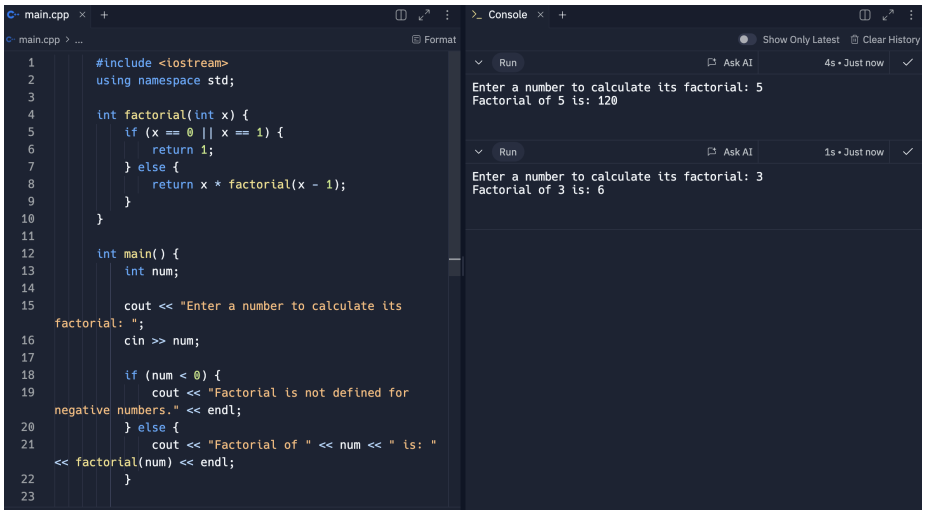

Figure 2: Output

5 Question 3

Question:

Write a program with recurrence function to calculate GCD (Greater Common Division)!

```cpp
#include <iostream>
using namespace std;

int gcd(int a, int b) {
    if (b == 0) {
        return a;
    } else {
        return gcd(b, a % b);
    }
}

int main() {
    int num1, num2;
    cout << "Enter two numbers to calculate their GCD: ";
    cin >> num1 >> num2;

    cout << "The GCD of " << num1 << " and " << num2 << " is: "
    ↪  << gcd(num1, num2) << endl;

    return 0;
```

```
}
```

6 Explanation

```
int gcd(int a, int b) {
    if (b == 0) {
        return a;
    } else {
        return gcd(b, a % b);
    }
}
```

This program calculates GCD using Euclid's algorithm:
- The GCD of two numbers a and b can be calculated using the relation gcd(a, b) = gcd(b, a % b), where % is the modulo operation.
- This continues until b becomes 0, at which point a will be the GCD.
- When b == 0, the GCD is a. The number stops at this point.
- If b is not 0, the function will use the arguments gcd(b, a % b). This reduces the problem to a simple form by applying the operation.

```
int main() {
    int num1, num2;
    cout << "Enter two numbers to calculate their GCD: ";
    cin >> num1 >> num2;

    cout << "The GCD of " << num1 << " and " << num2 << " is: "
    ↪  << gcd(num1, num2) << endl;

    return 0;
}
```

- The program asks the user to enter two number, num1 and num2, using cin >> num1 >> num2;.
- The function gcd(num1, num2) is used to compute the GCD of the two numbers inputted by the user.

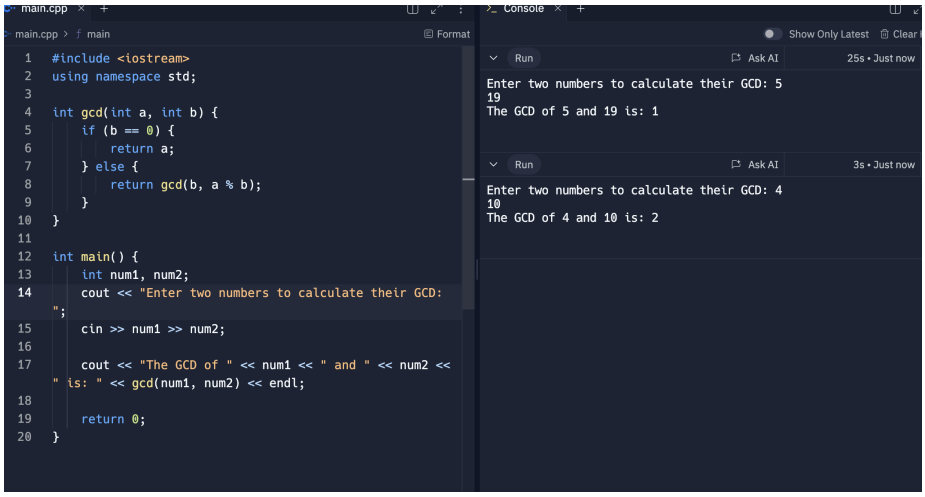

Figure 3: Test Output