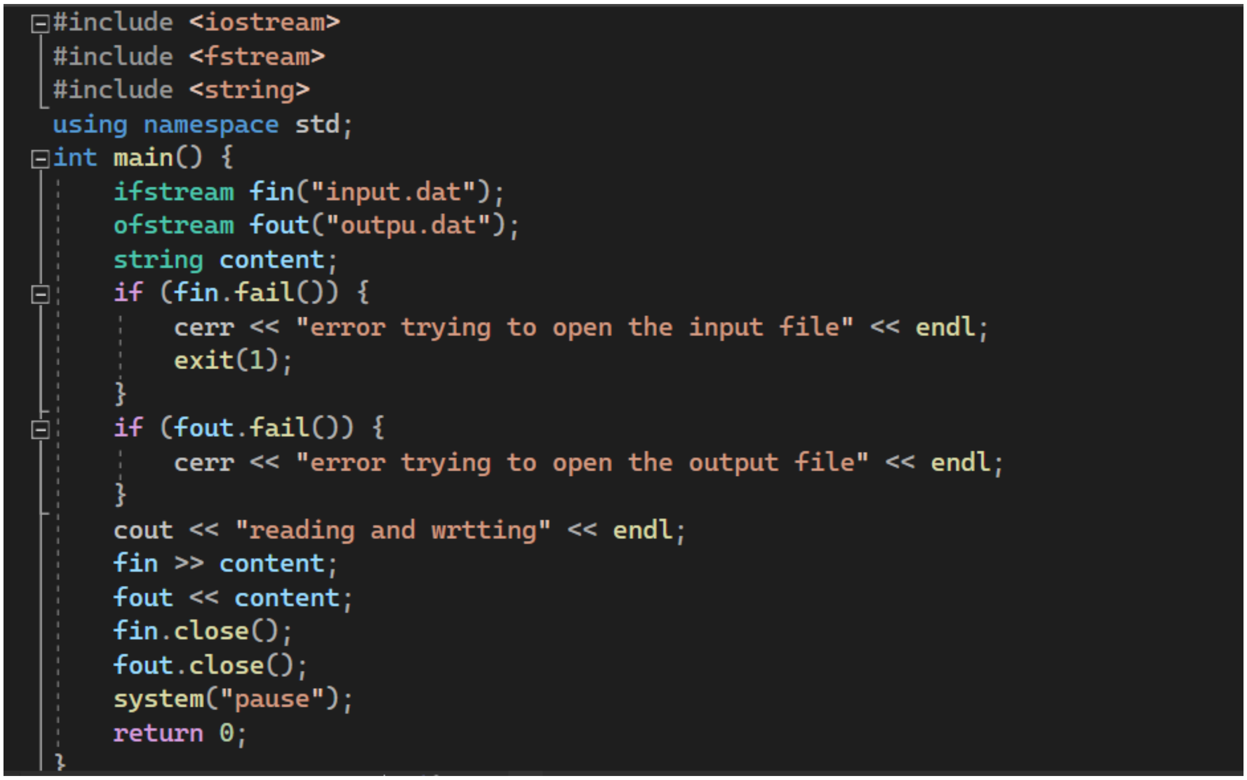
Naafiul Hossain
ESE224
115107623
Tuesday 10-12:50am

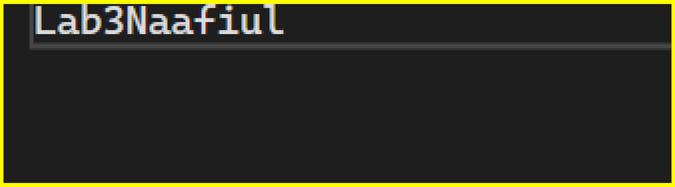
Problem 1:

Main.cpp



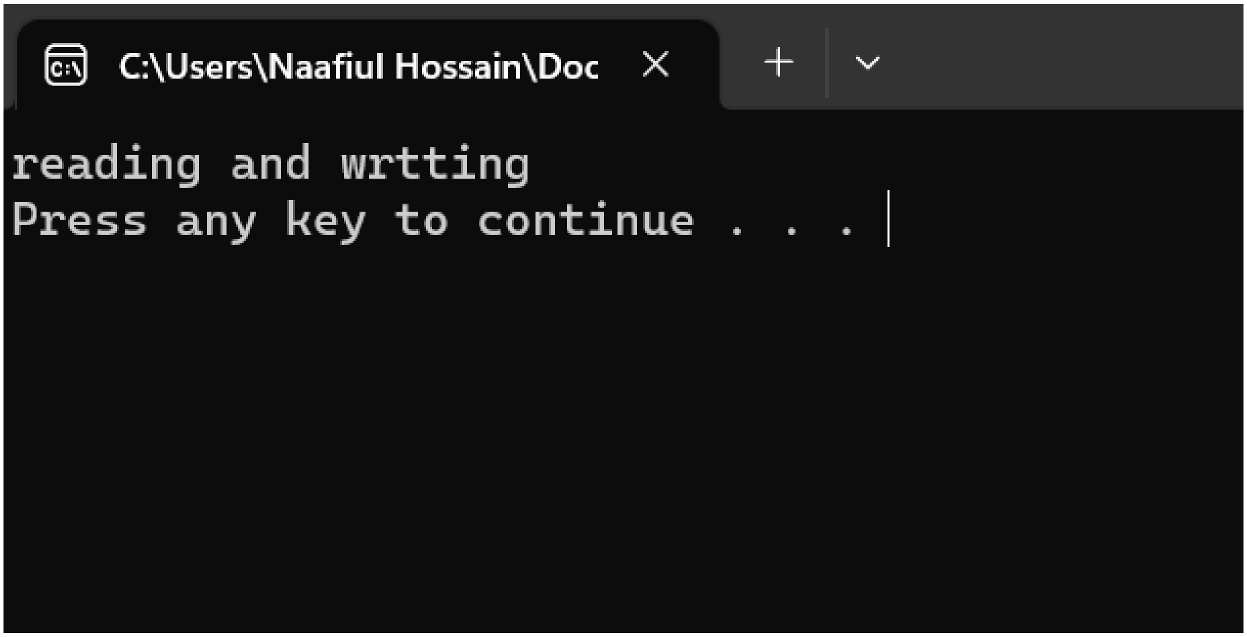
```
#include <iostream>
#include <fstream>
#include <string>
using namespace std;
int main() {
    ifstream fin("input.dat");
    ofstream fout("outpu.dat");
    string content;
    if (fin.fail()) {
        cerr << "error trying to open the input file" << endl;
        exit(1);
    }
    if (fout.fail()) {
        cerr << "error trying to open the output file" << endl;
    }
    cout << "reading and wrtting" << endl;
    fin >> content;
    fout << content;
    fin.close();
    fout.close();
    system("pause");
    return 0;
}
```

Input.dat



```
Lab3Naafiul
```

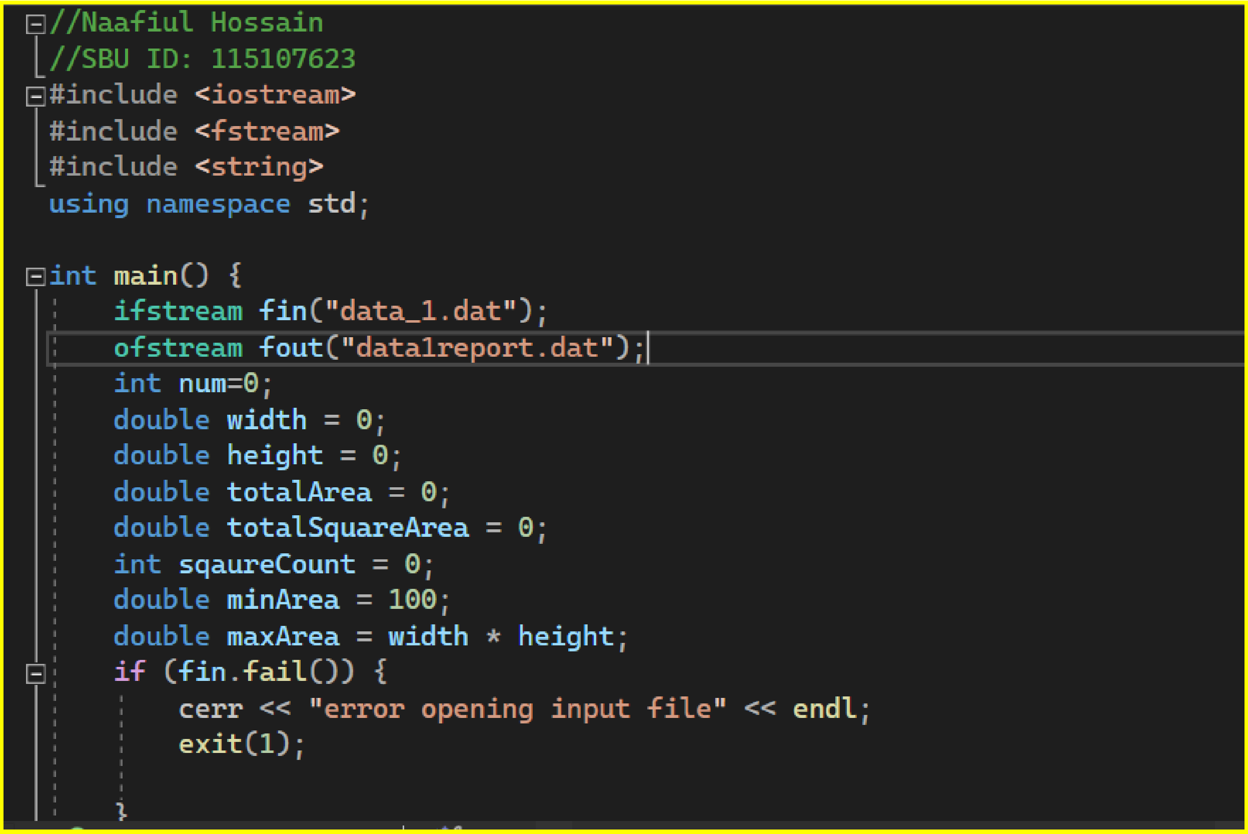
Screenshot of the running program:



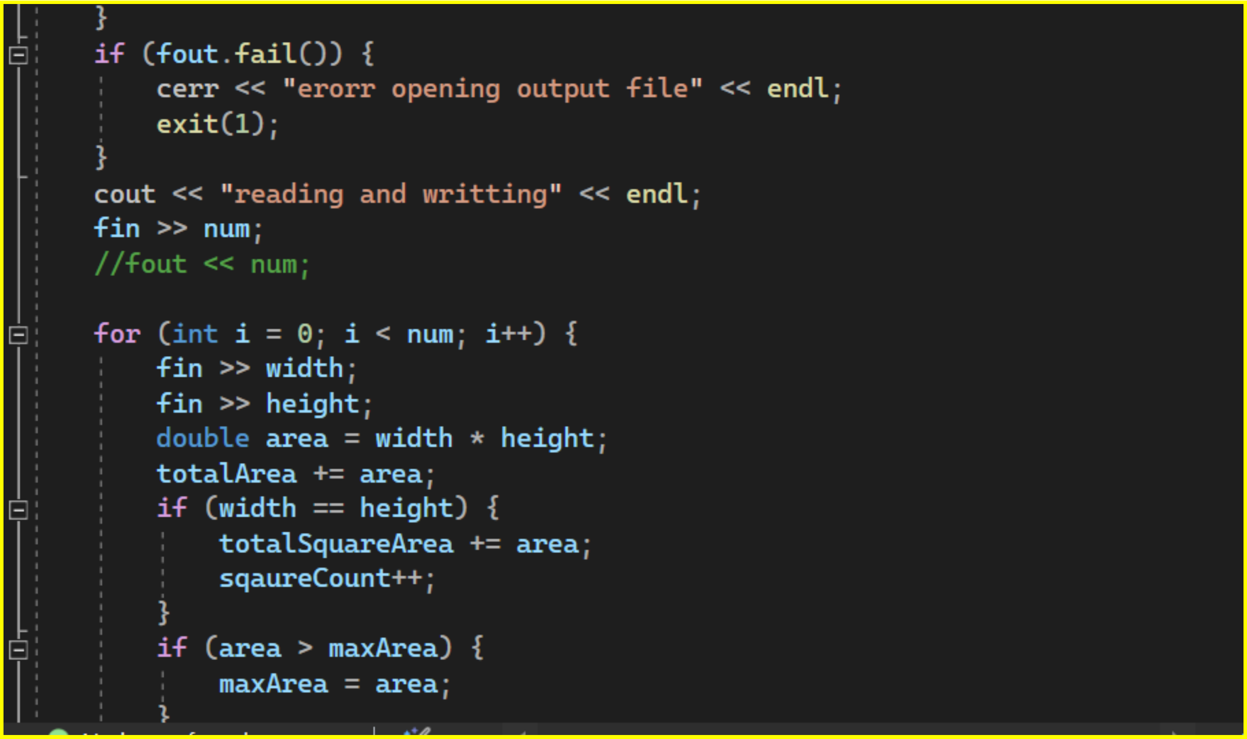
```
reading and wrtting
Press any key to continue . . .
```

Problem 2

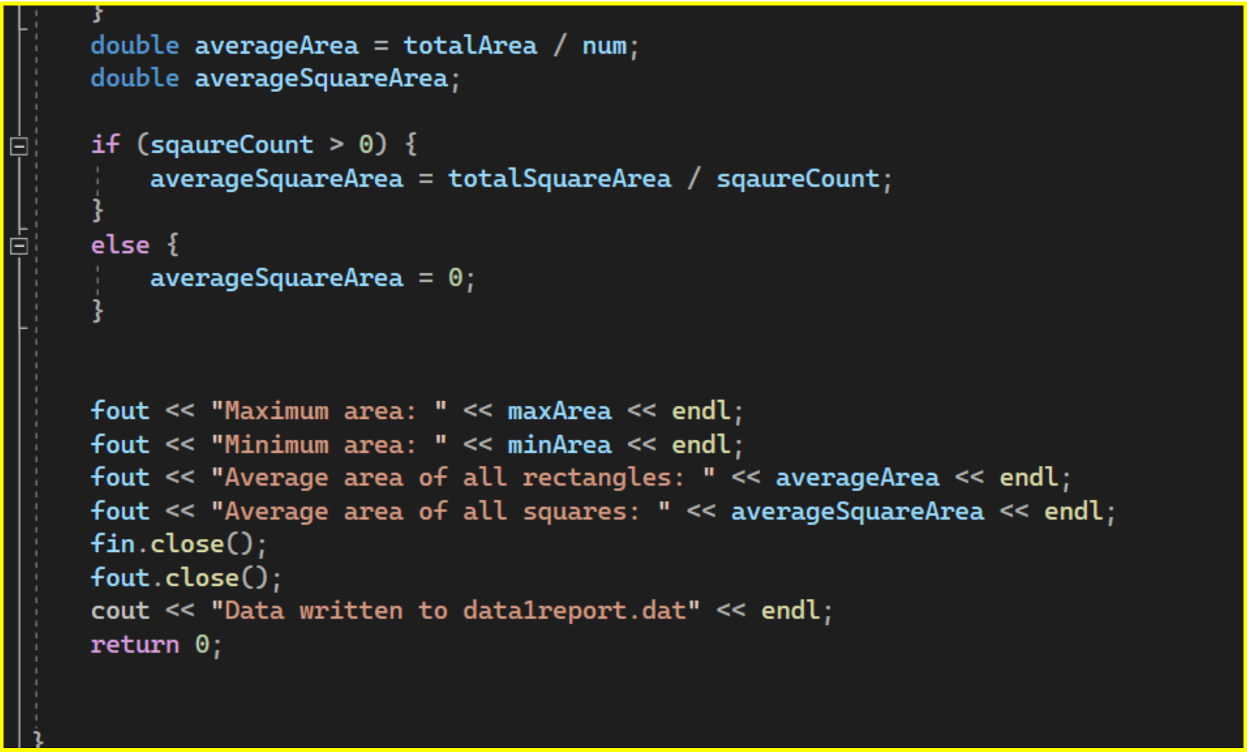
main.h



```cpp
//Naafiul Hossain
//SBU ID: 115107623
#include <iostream>
#include <fstream>
#include <string>
using namespace std;

int main() {
    ifstream fin("data_1.dat");
    ofstream fout("data1report.dat");
    int num=0;
    double width = 0;
    double height = 0;
    double totalArea = 0;
    double totalSquareArea = 0;
    int sqaureCount = 0;
    double minArea = 100;
    double maxArea = width * height;
    if (fin.fail()) {
        cerr << "error opening input file" << endl;
        exit(1);

    }
```

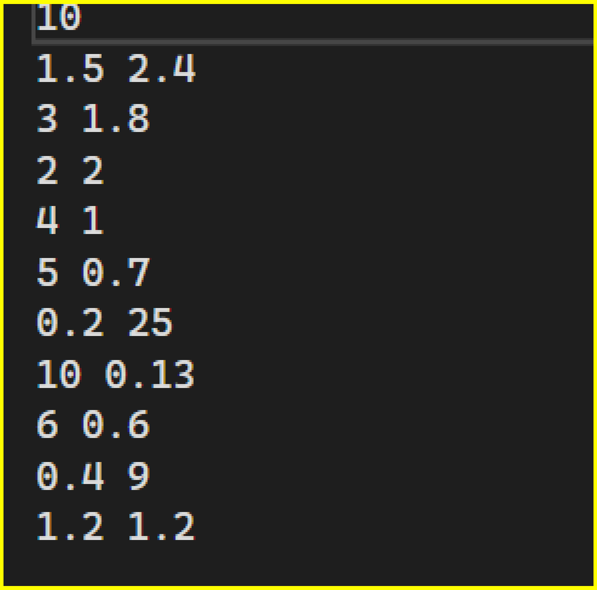


```
    }
    if (fout.fail()) {
        cerr << "erorr opening output file" << endl;
        exit(1);
    }
    cout << "reading and writting" << endl;
    fin >> num;
    //fout << num;

    for (int i = 0; i < num; i++) {
        fin >> width;
        fin >> height;
        double area = width * height;
        totalArea += area;
        if (width == height) {
            totalSquareArea += area;
            sqaureCount++;
        }
        if (area > maxArea) {
            maxArea = area;
        }
```



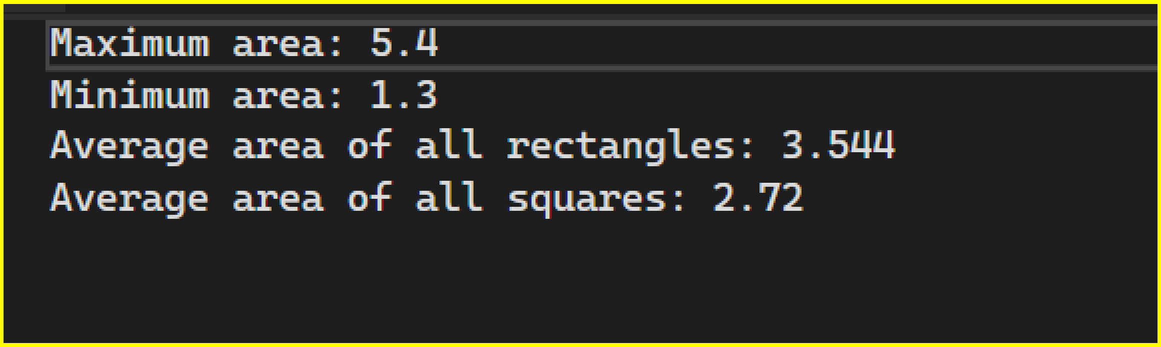
```
    }
    double averageArea = totalArea / num;
    double averageSquareArea;

    if (sqaureCount > 0) {
        averageSquareArea = totalSquareArea / sqaureCount;
    }
    else {
        averageSquareArea = 0;
    }


    fout << "Maximum area: " << maxArea << endl;
    fout << "Minimum area: " << minArea << endl;
    fout << "Average area of all rectangles: " << averageArea << endl;
    fout << "Average area of all squares: " << averageSquareArea << endl;
    fin.close();
    fout.close();
    cout << "Data written to data1report.dat" << endl;
    return 0;


}
```

Data1_.dat:



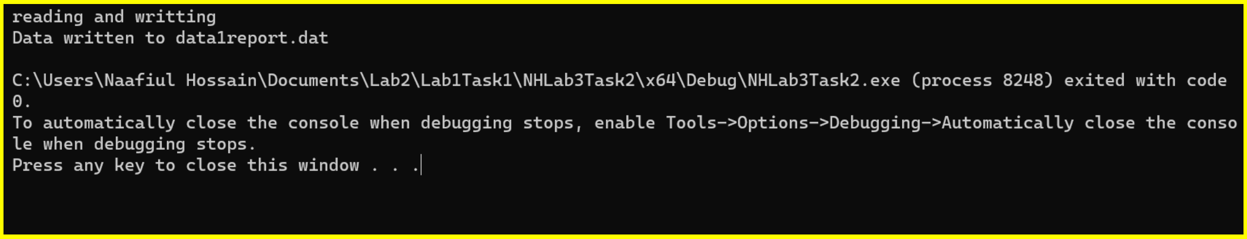
```
10
1.5 2.4
3 1.8
2 2
4 1
5 0.7
0.2 25
10 0.13
6 0.6
0.4 9
1.2 1.2
```

Data1report.dat



```
Maximum area: 5.4
Minimum area: 1.3
Average area of all rectangles: 3.544
Average area of all squares: 2.72
```

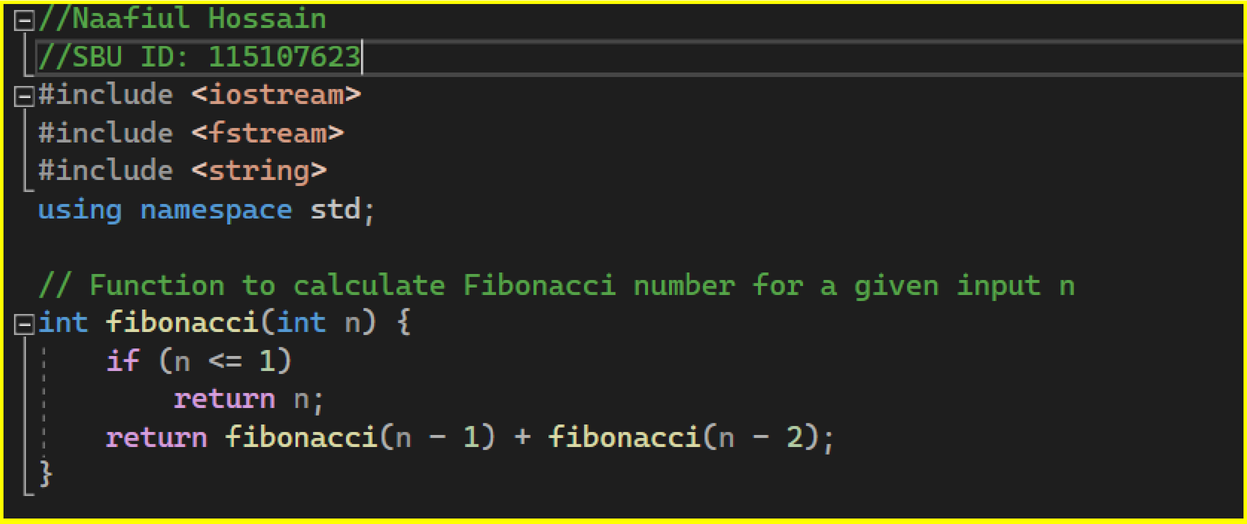
Running of the Program:



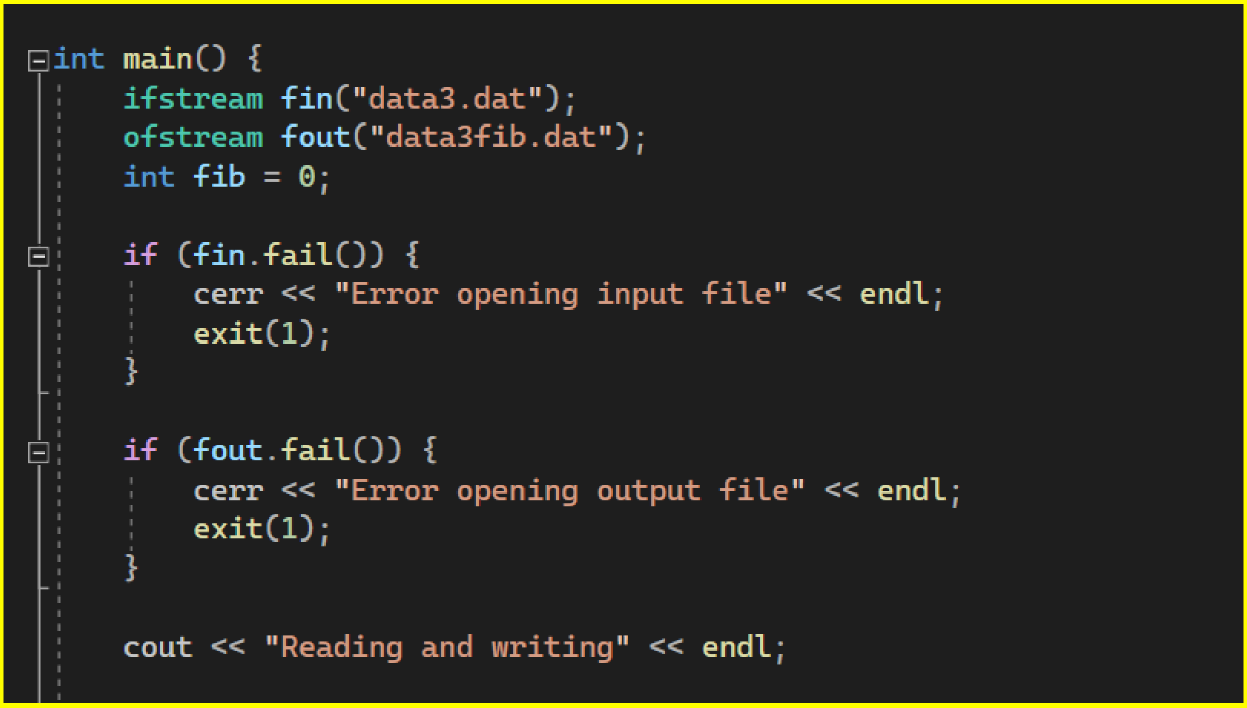
```
reading and writting
Data written to data1report.dat

C:\Users\Naafiul Hossain\Documents\Lab2\Lab1Task1\NHLab3Task2\x64\Debug\NHLab3Task2.exe (process 8248) exited with code
0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the conso
le when debugging stops.
Press any key to close this window . . .
```

Problem 3

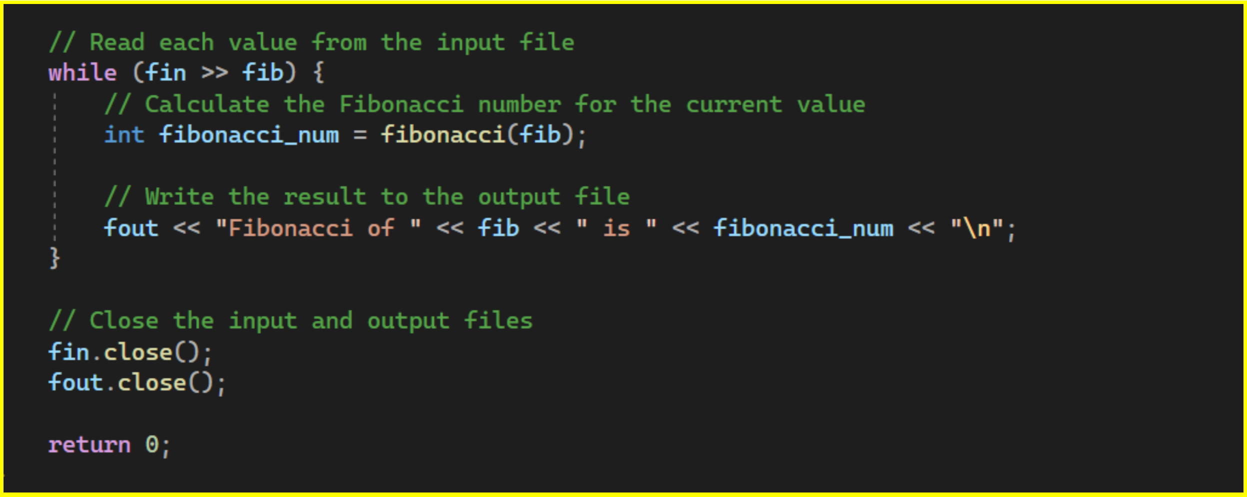
Main.cpp



```cpp
//Naafiul Hossain
//SBU ID: 115107623
#include <iostream>
#include <fstream>
#include <string>
using namespace std;

// Function to calculate Fibonacci number for a given input n
int fibonacci(int n) {
    if (n <= 1)
        return n;
    return fibonacci(n - 1) + fibonacci(n - 2);
}
```

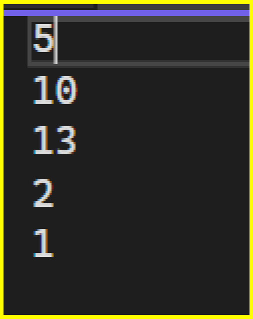


```cpp
int main() {
    ifstream fin("data3.dat");
    ofstream fout("data3fib.dat");
    int fib = 0;

    if (fin.fail()) {
        cerr << "Error opening input file" << endl;
        exit(1);
    }

    if (fout.fail()) {
        cerr << "Error opening output file" << endl;
        exit(1);
    }

    cout << "Reading and writing" << endl;
```



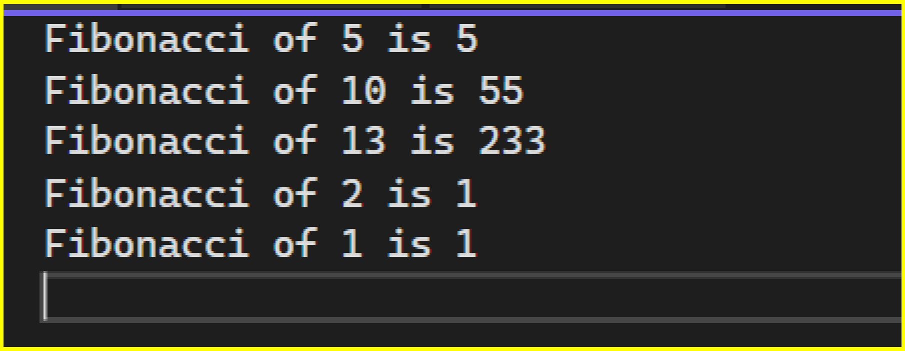
```
// Read each value from the input file
while (fin >> fib) {
    // Calculate the Fibonacci number for the current value
    int fibonacci_num = fibonacci(fib);

    // Write the result to the output file
    fout << "Fibonacci of " << fib << " is " << fibonacci_num << "\n";
}

// Close the input and output files
fin.close();
fout.close();

return 0;
```

Data3.dat



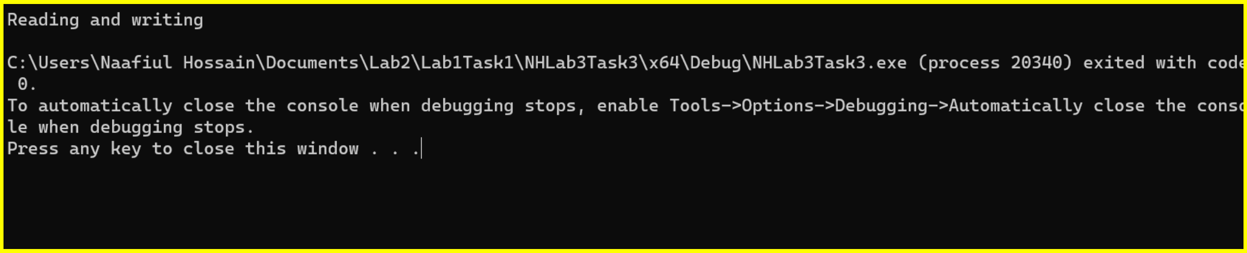
```
5
10
13
2
1
```

Data3fib,dat:



```
Fibonacci of 5 is 5
Fibonacci of 10 is 55
Fibonacci of 13 is 233
Fibonacci of 2 is 1
Fibonacci of 1 is 1
```

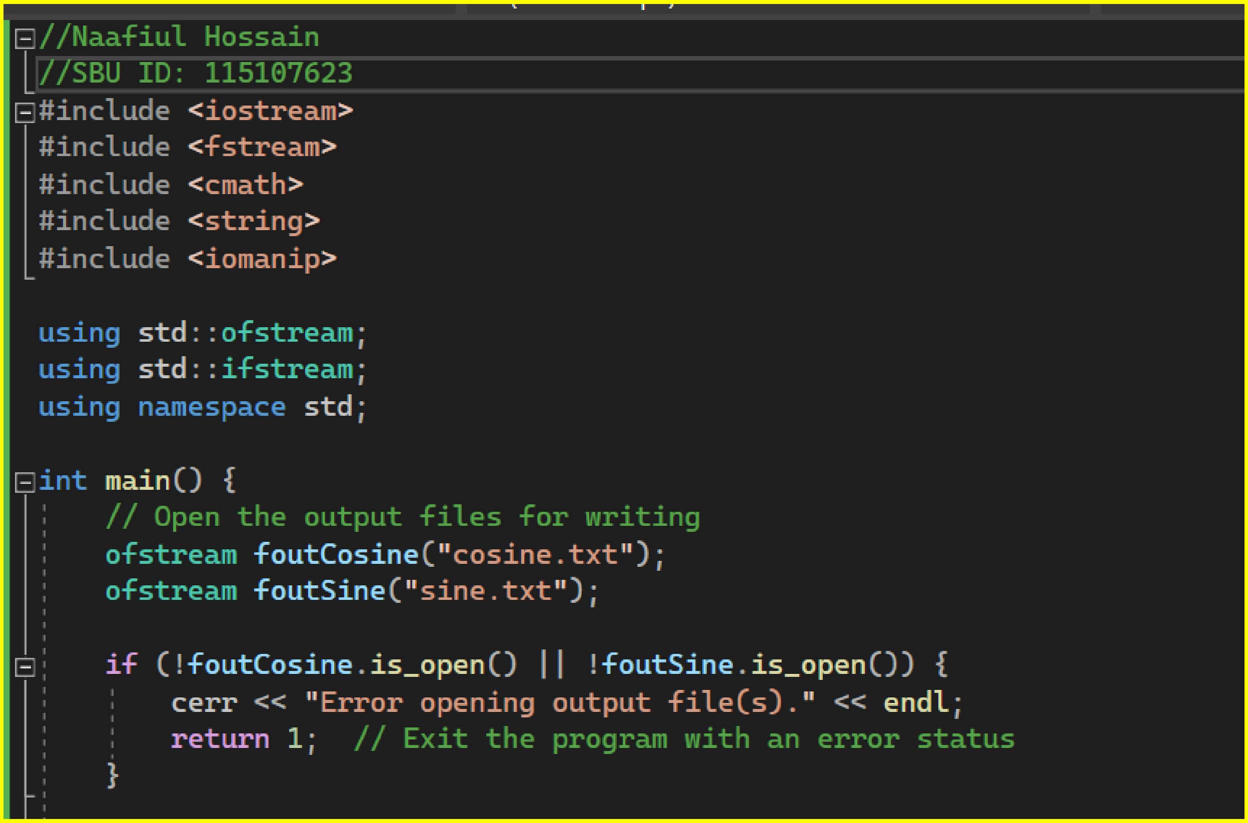
Screenshot of the running program:



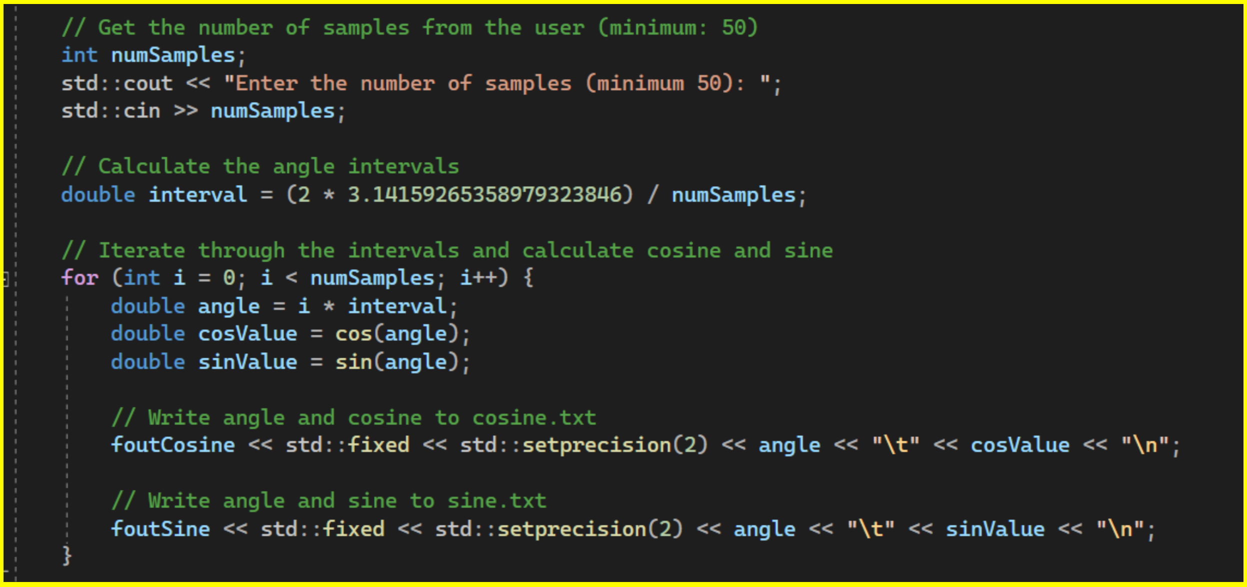
```
Reading and writing

C:\Users\Naafiul Hossain\Documents\Lab2\Lab1Task1\NHLab3Task3\x64\Debug\NHLab3Task3.exe (process 20340) exited with code
 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the conso
le when debugging stops.
Press any key to close this window . . .
```

Problem 4

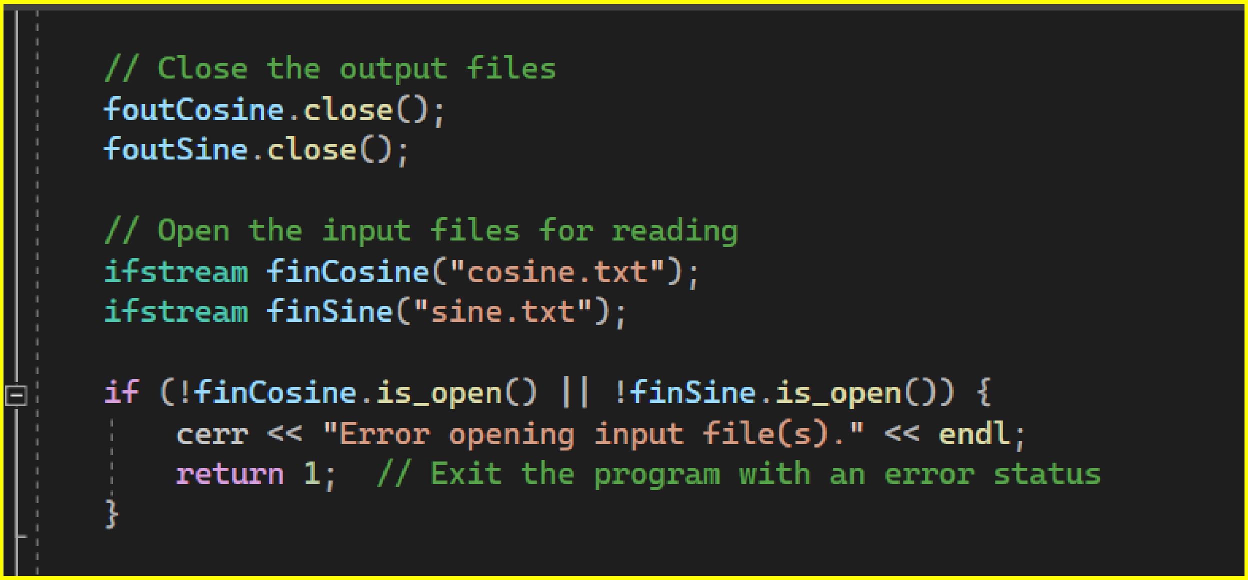
Main.cpp



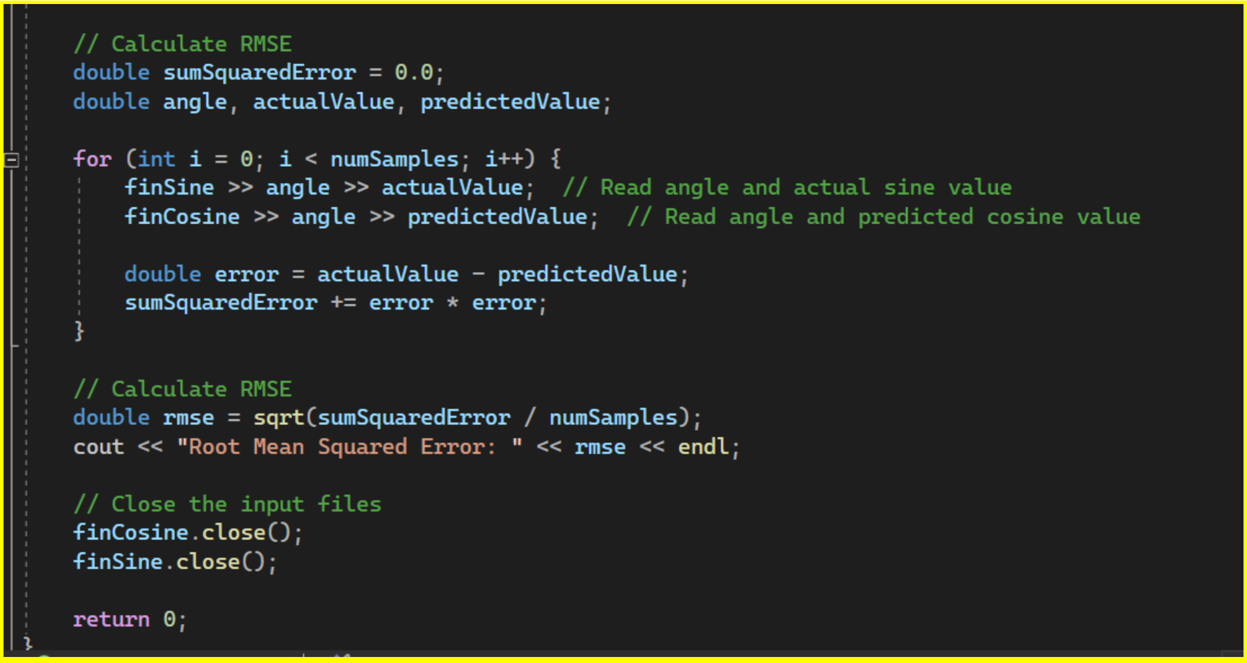
```cpp
//Naafiul Hossain
//SBU ID: 115107623
#include <iostream>
#include <fstream>
#include <cmath>
#include <string>
#include <iomanip>

using std::ofstream;
using std::ifstream;
using namespace std;

int main() {
    // Open the output files for writing
    ofstream foutCosine("cosine.txt");
    ofstream foutSine("sine.txt");

    if (!foutCosine.is_open() || !foutSine.is_open()) {
        cerr << "Error opening output file(s)." << endl;
        return 1;  // Exit the program with an error status
    }
```



```cpp
    // Get the number of samples from the user (minimum: 50)
    int numSamples;
    std::cout << "Enter the number of samples (minimum 50): ";
    std::cin >> numSamples;

    // Calculate the angle intervals
    double interval = (2 * 3.14159265358979323846) / numSamples;

    // Iterate through the intervals and calculate cosine and sine
    for (int i = 0; i < numSamples; i++) {
        double angle = i * interval;
        double cosValue = cos(angle);
        double sinValue = sin(angle);

        // Write angle and cosine to cosine.txt
        foutCosine << std::fixed << std::setprecision(2) << angle << "\t" << cosValue << "\n";

        // Write angle and sine to sine.txt
        foutSine << std::fixed << std::setprecision(2) << angle << "\t" << sinValue << "\n";
    }
```

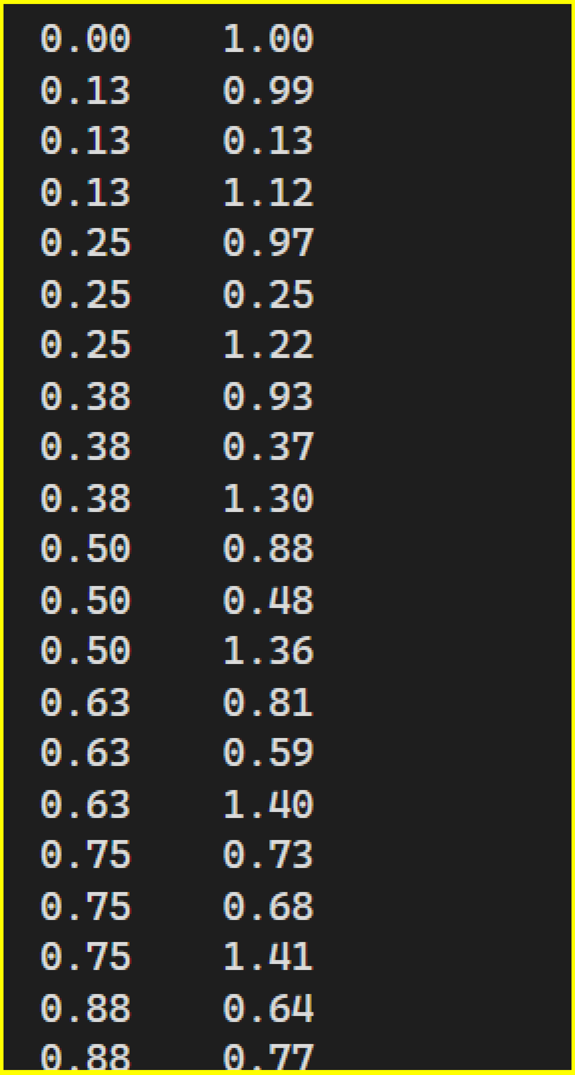


```
    // Close the output files
    foutCosine.close();
    foutSine.close();

    // Open the input files for reading
    ifstream finCosine("cosine.txt");
    ifstream finSine("sine.txt");

    if (!finCosine.is_open() || !finSine.is_open()) {
        cerr << "Error opening input file(s)." << endl;
        return 1;  // Exit the program with an error status
    }
```



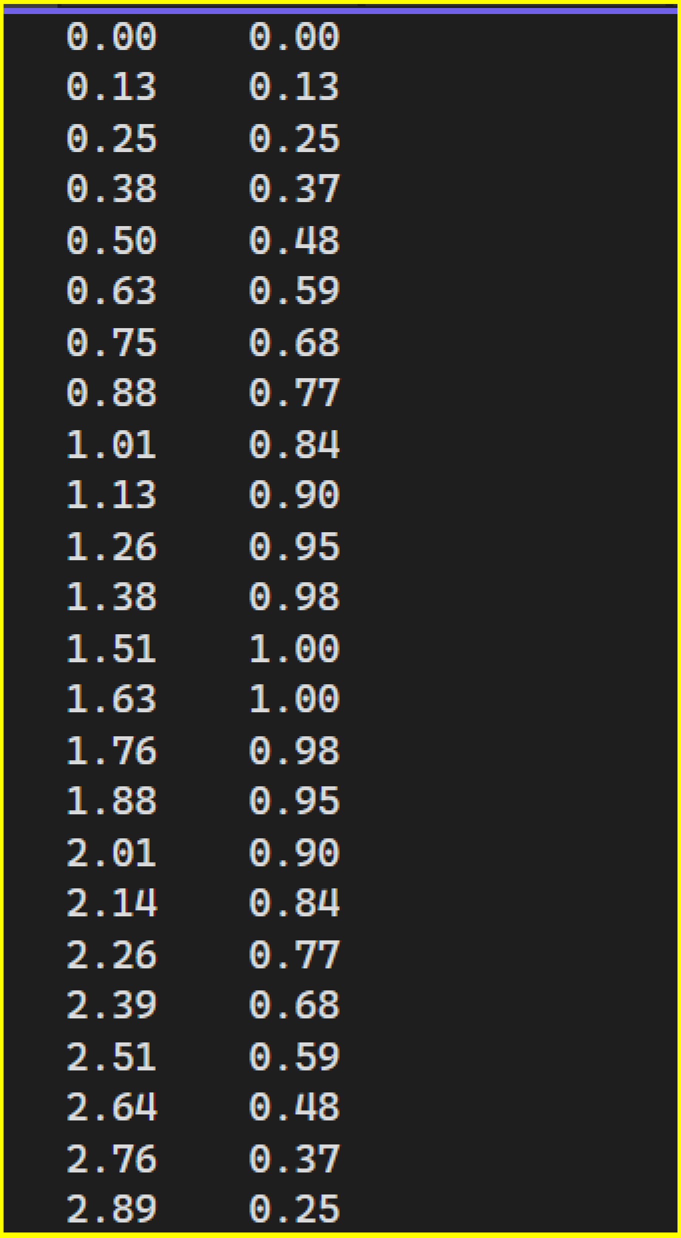
```
    // Calculate RMSE
    double sumSquaredError = 0.0;
    double angle, actualValue, predictedValue;

    for (int i = 0; i < numSamples; i++) {
        finSine >> angle >> actualValue;  // Read angle and actual sine value
        finCosine >> angle >> predictedValue;  // Read angle and predicted cosine value

        double error = actualValue - predictedValue;
        sumSquaredError += error * error;
    }

    // Calculate RMSE
    double rmse = sqrt(sumSquaredError / numSamples);
    cout << "Root Mean Squared Error: " << rmse << endl;

    // Close the input files
    finCosine.close();
    finSine.close();

    return 0;
}
```

Result.txt:



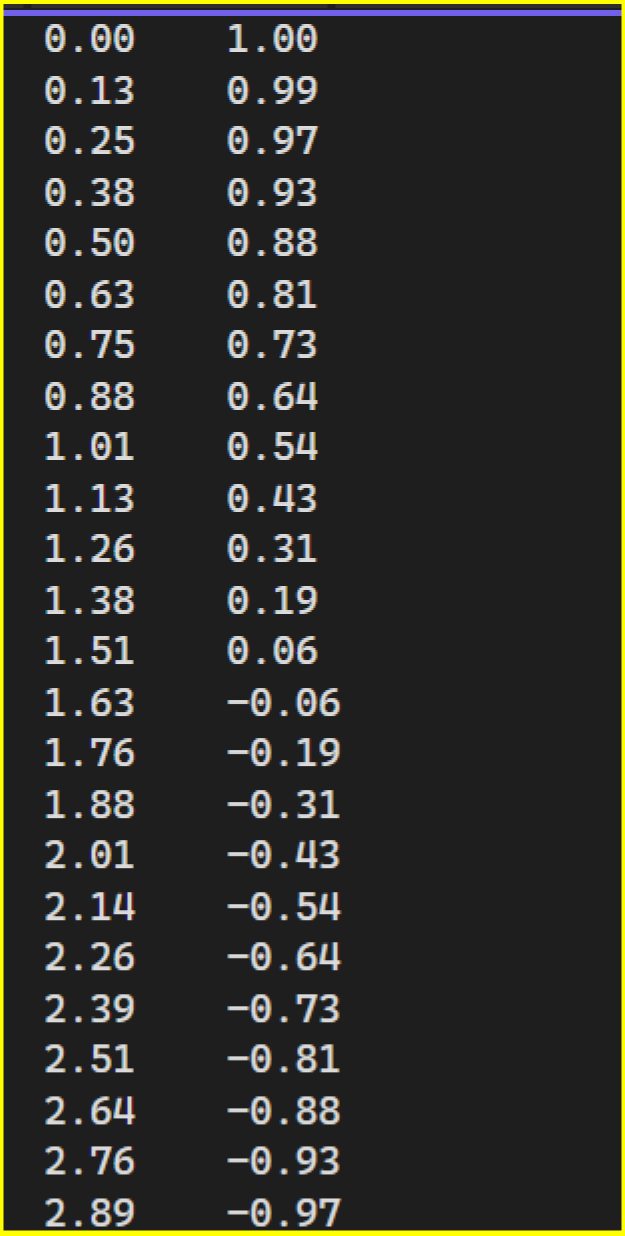
```
0.00    1.00
0.13    0.99
0.13    0.13
0.13    1.12
0.25    0.97
0.25    0.25
0.25    1.22
0.38    0.93
0.38    0.37
0.38    1.30
0.50    0.88
0.50    0.48
0.50    1.36
0.63    0.81
0.63    0.59
0.63    1.40
0.75    0.73
0.75    0.68
0.75    1.41
0.88    0.64
0.88    0.77
```

Sine.txt:



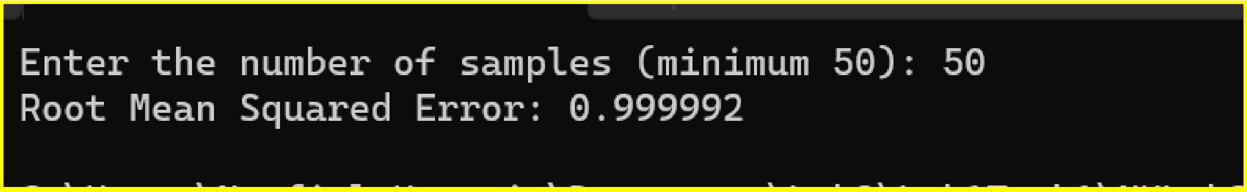
```
   0.00     0.00
   0.13     0.13
   0.25     0.25
   0.38     0.37
   0.50     0.48
   0.63     0.59
   0.75     0.68
   0.88     0.77
   1.01     0.84
   1.13     0.90
   1.26     0.95
   1.38     0.98
   1.51     1.00
   1.63     1.00
   1.76     0.98
   1.88     0.95
   2.01     0.90
   2.14     0.84
   2.26     0.77
   2.39     0.68
   2.51     0.59
   2.64     0.48
   2.76     0.37
   2.89     0.25
```

Cosine.txt



```
0.00    1.00
0.13    0.99
0.25    0.97
0.38    0.93
0.50    0.88
0.63    0.81
0.75    0.73
0.88    0.64
1.01    0.54
1.13    0.43
1.26    0.31
1.38    0.19
1.51    0.06
1.63    -0.06
1.76    -0.19
1.88    -0.31
2.01    -0.43
2.14    -0.54
2.26    -0.64
2.39    -0.73
2.51    -0.81
2.64    -0.88
2.76    -0.93
2.89    -0.97
```

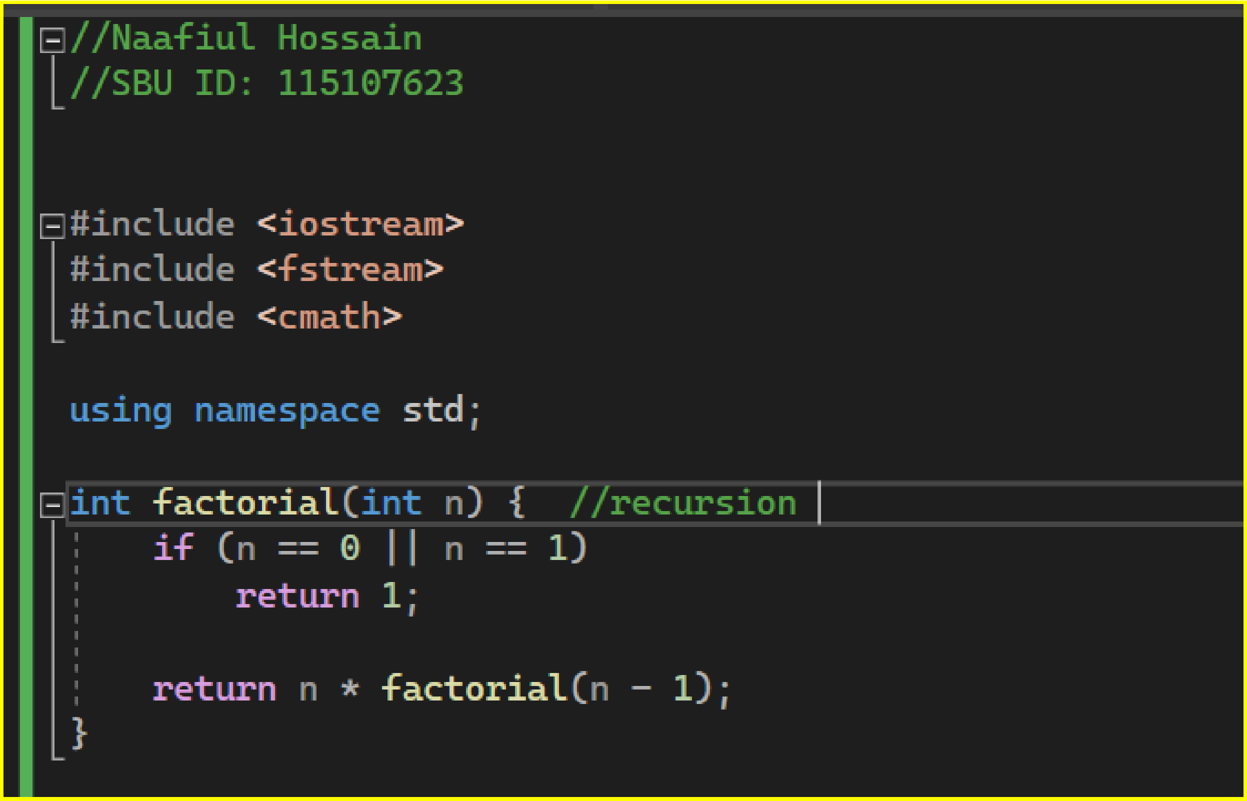
Screenshot of the Program running:



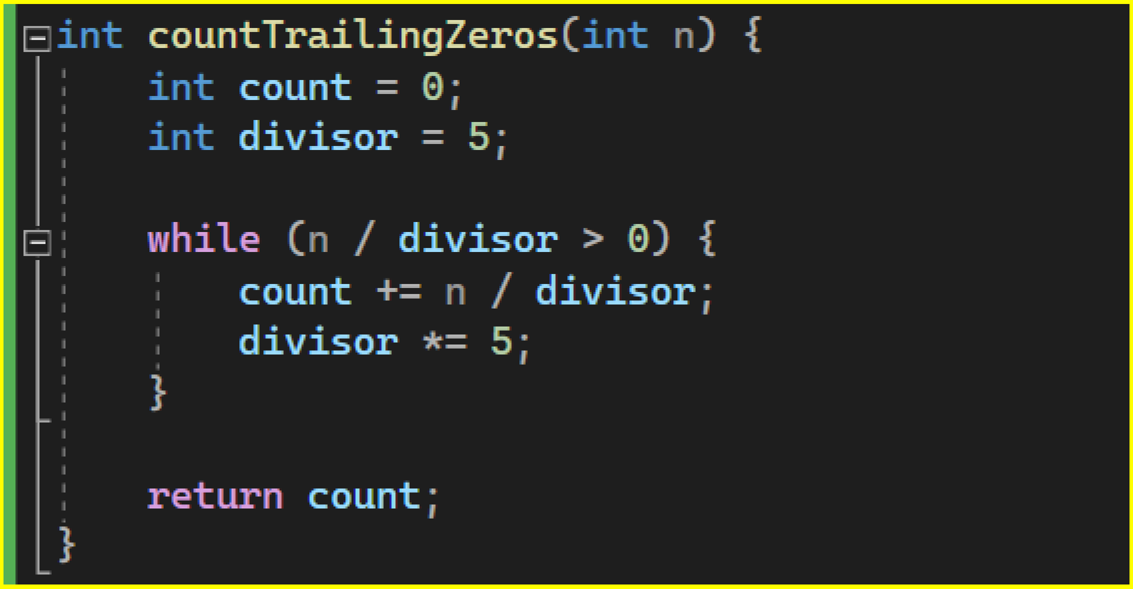
```
Enter the number of samples (minimum 50): 50
Root Mean Squared Error: 0.999992
```

Problem 5

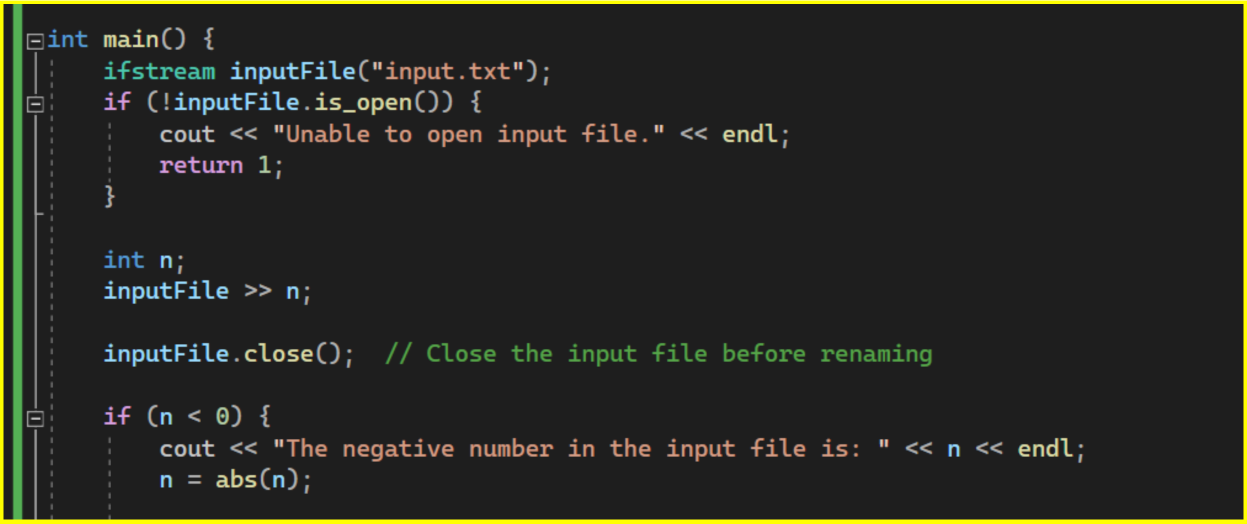
Main.cpp



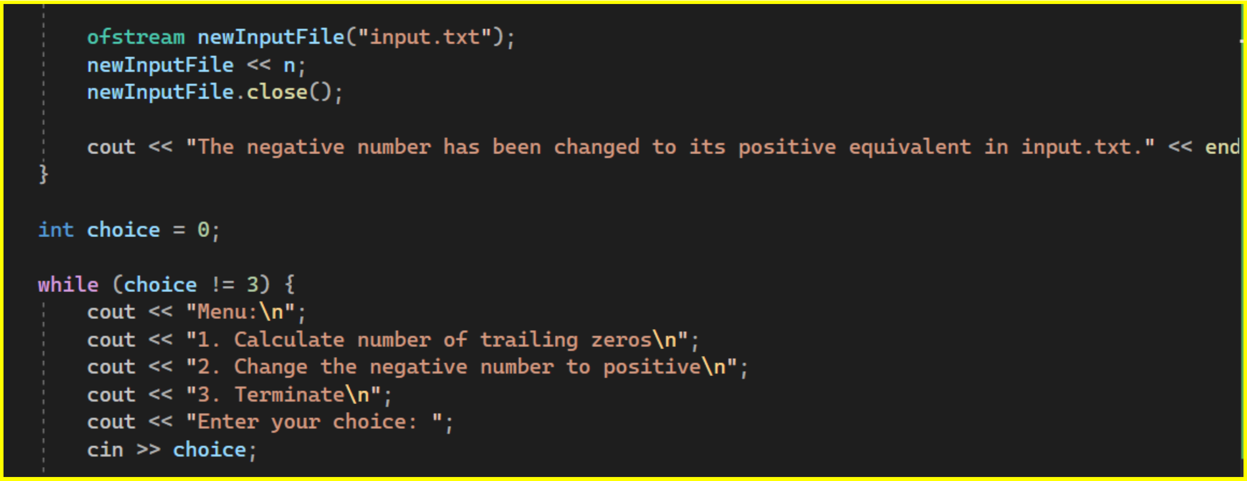
```
//Naafiul Hossain
//SBU ID: 115107623


#include <iostream>
#include <fstream>
#include <cmath>

using namespace std;

int factorial(int n) {  //recursion
    if (n == 0 || n == 1)
        return 1;

    return n * factorial(n - 1);
}
```



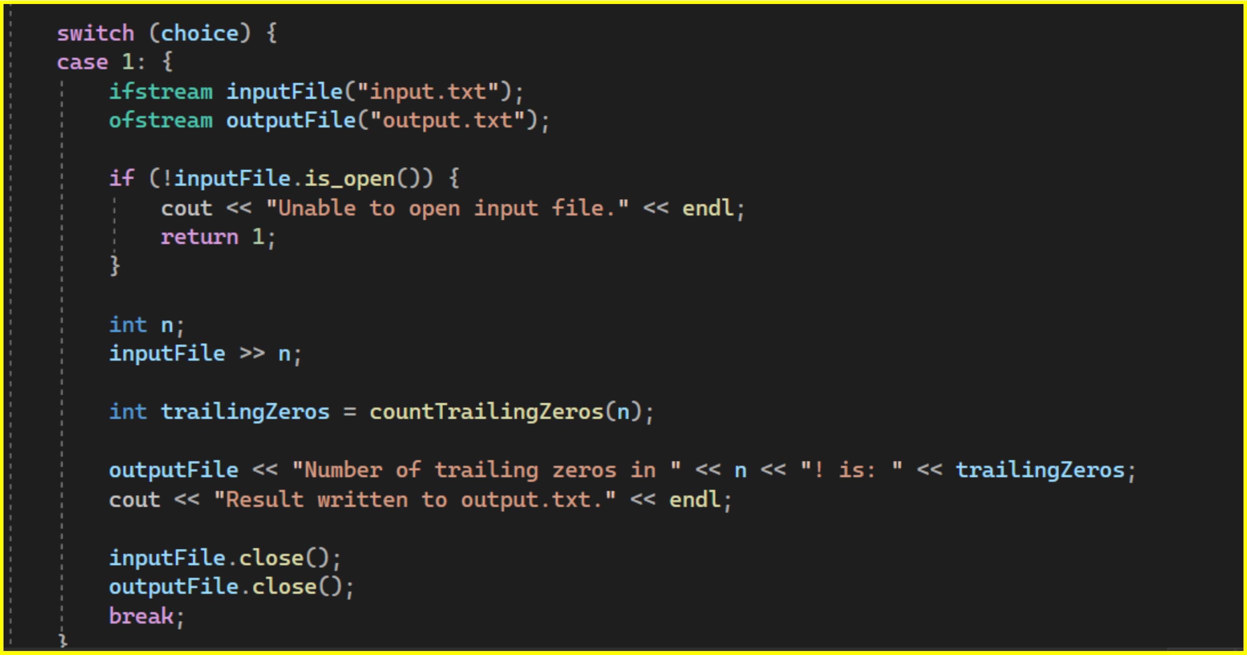
```
int countTrailingZeros(int n) {
    int count = 0;
    int divisor = 5;

    while (n / divisor > 0) {
        count += n / divisor;
        divisor *= 5;
    }

    return count;
}
```



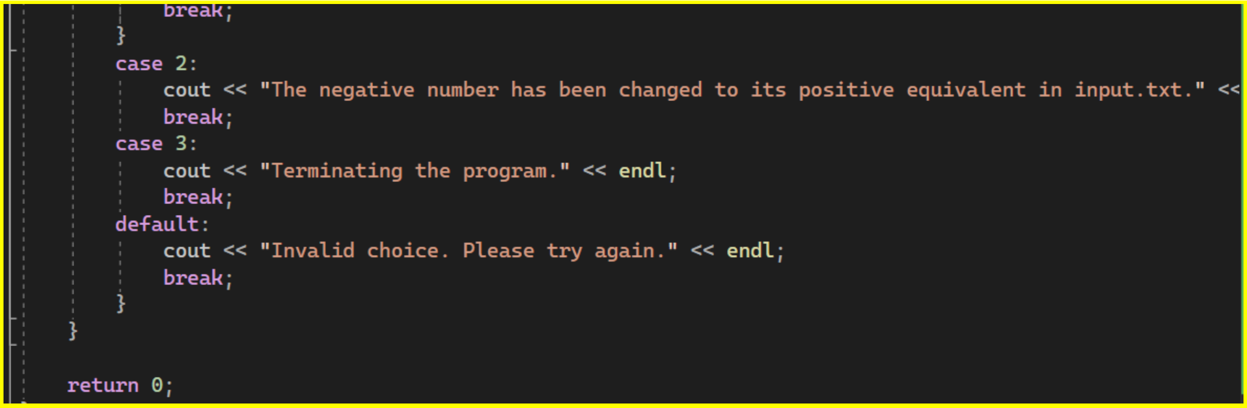
```
int main() {
    ifstream inputFile("input.txt");
    if (!inputFile.is_open()) {
        cout << "Unable to open input file." << endl;
        return 1;
    }

    int n;
    inputFile >> n;

    inputFile.close();  // Close the input file before renaming

    if (n < 0) {
        cout << "The negative number in the input file is: " << n << endl;
        n = abs(n);
```



```
        ofstream newInputFile("input.txt");
        newInputFile << n;
        newInputFile.close();

        cout << "The negative number has been changed to its positive equivalent in input.txt." << end
    }

    int choice = 0;

    while (choice != 3) {
        cout << "Menu:\n";
        cout << "1. Calculate number of trailing zeros\n";
        cout << "2. Change the negative number to positive\n";
        cout << "3. Terminate\n";
        cout << "Enter your choice: ";
        cin >> choice;
```

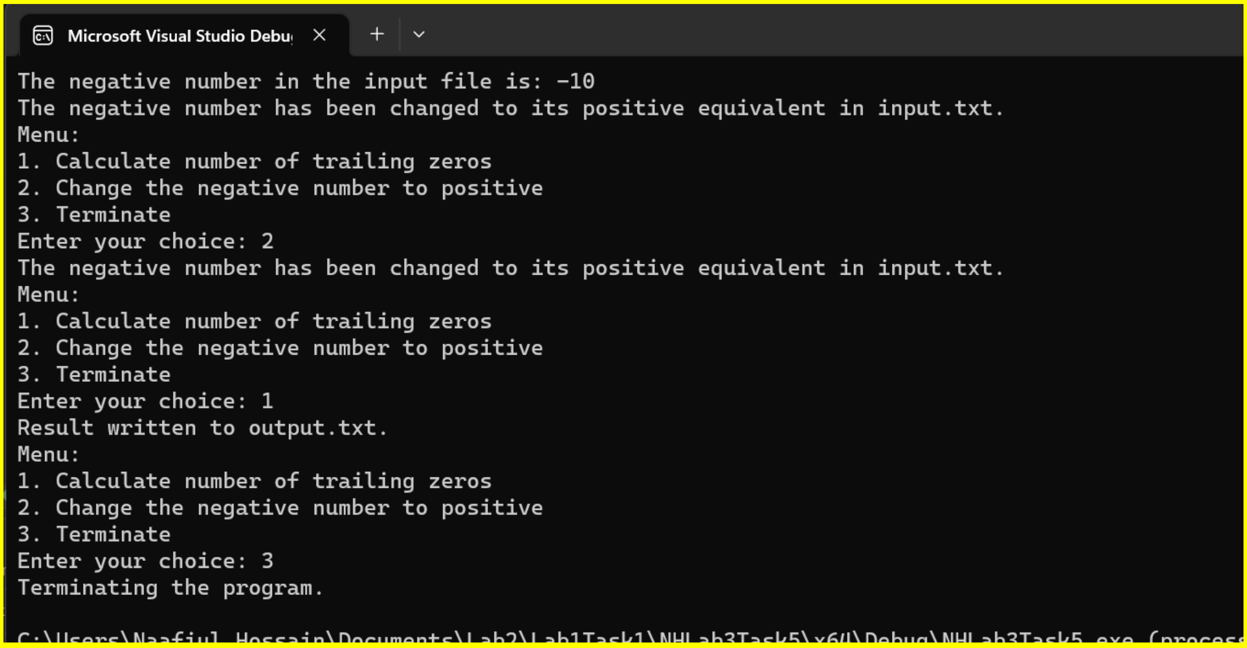


```
        switch (choice) {
        case 1: {
            ifstream inputFile("input.txt");
            ofstream outputFile("output.txt");

            if (!inputFile.is_open()) {
                cout << "Unable to open input file." << endl;
                return 1;
            }

            int n;
            inputFile >> n;

            int trailingZeros = countTrailingZeros(n);

            outputFile << "Number of trailing zeros in " << n << "! is: " << trailingZeros;
            cout << "Result written to output.txt." << endl;

            inputFile.close();
            outputFile.close();
            break;
        }
```



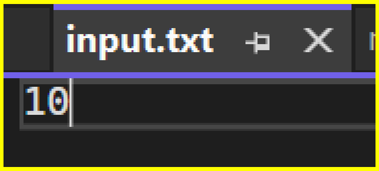
```
            break;
        }
        case 2:
            cout << "The negative number has been changed to its positive equivalent in input.txt." <<
            break;
        case 3:
            cout << "Terminating the program." << endl;
            break;
        default:
            cout << "Invalid choice. Please try again." << endl;
            break;
        }
    }

    return 0;
```

Screenshot of the running program:



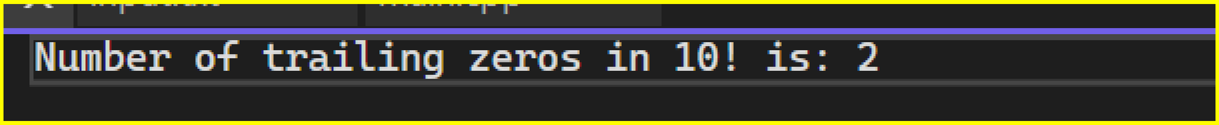
```
The negative number in the input file is: -10
The negative number has been changed to its positive equivalent in input.txt.
Menu:
1. Calculate number of trailing zeros
2. Change the negative number to positive
3. Terminate
Enter your choice: 2
The negative number has been changed to its positive equivalent in input.txt.
Menu:
1. Calculate number of trailing zeros
2. Change the negative number to positive
3. Terminate
Enter your choice: 1
Result written to output.txt.
Menu:
1. Calculate number of trailing zeros
2. Change the negative number to positive
3. Terminate
Enter your choice: 3
Terminating the program.

C:\Users\Naafiul Hossain\Documents\Lab2\Lab1Task1\NHLab3Task5\x64\Debug\NHLab3Task5.exe (process
```

Input.txt



```
10
```

Output.txt



```
Number of trailing zeros in 10! is: 2
```