Naafiul Hossain
ESE224
115107623
Tuesday 10-12:50am

Problem 1:

Point.h

```
#pragma once
class Point {
private:
    double xCord, yCord;
public:
    Point();
    Point(double x, double y);
    double getX();
    double getY();
    void setX(double x);
    void setY(double y);
    double dist2origin();
    double Distance(const Point& p2) const;
    double operator -(const Point& p2) const;
    bool Equal(const Point& p2) const;
    bool operator==(const Point& p2)const;
    void Print();
    void operator*(double n);
    bool operator>(const Point& p2) const;


};
```

Point.cpp

```cpp
double Point::operator-(const Point& p2) const {
    double dx = p2.xCord - xCord;
    double dy = p2.yCord - yCord;
    return sqrt(pow(dx, 2) + pow(dy, 2));
}
bool Point::Equal(const Point& p2) const {
    return (p2.xCord == xCord) && (p2.yCord == yCord);

}
bool Point::operator==(const Point& p2) const {
    return (p2.xCord == xCord) && (p2.yCord == yCord);
}
void Point::Print() {
    cout.setf(ios::fixed);
    cout.precision(3);
    cout << "The Point is (" << xCord << "," << yCord << ")" << endl;
}
void Point::operator*(double n) {
    xCord = xCord * n;
    yCord = yCord * n;
}
bool Point::operator>(const Point& p2)const {
    double d1 = sqrt(pow(xCord, 2) + pow(yCord, 2));
```

```cpp
}
bool Point::operator>(const Point& p2)const {
    double d1 = sqrt(pow(xCord, 2) + pow(yCord, 2));
    double d2= sqrt(pow(p2.xCord, 2) + pow(p2.yCord, 2));
    return d1 > d2;
}
```

Main.cpp

```
    double x, y, r;
    cout << "Enter the cords of p1:" << endl;
    cin >> x >> y;
    Point p1(x, y);
    p1.Print();
    cout << "Enter the cords of p2:" << endl;
    cin >> x >> y;
    Point p2(x, y);
    p2.Print();
    cout << "The distance between two points is" << p1.Distance(p2) << endl;
    cout << "The distance between two points is" << p1-p2 << endl;
    cout << "Are the two points the same? The answer is " << p2.Equal(p1) << endl;
    cout << "Are the two points the same The answer is" << (p1==p2) << endl<<endl;
    cout << "Enter a number to change the coords of p2: " << endl;
    cin >> r;
    p2* r;
    p2.Print();
    cout << "p1 > p2 ?" << (p1 > p2) << endl;
    system("pause");
    return 0;
}
```

Screenshot of the running program:

```
Enter the cords of p1:
1 2
The Point is (1.000,2.000)
Enter the cords of p2:
2 3
The Point is (2.000,3.000)
The distance between two points is1.414
The distance between two points is1.414
Are the two points the same? The answer is 0
Are the two points the same The answer is0

Enter a number to change the coords of p2:
3 2
The Point is (6.000,9.000)
p1 > p2 ?0
Press any key to continue . . .
```

Problem 2

Professor informed us this exercise is no longer required.

Problem 3

Pyramid.h

```h
//Naafiul Hossain
//SBU ID: 115107623
#pragma once
#ifndef PYRAMID_H
#define PYRAMID_H

class Pyramid {
private:
    int height;

public:
    Pyramid(int height);
    void create();
    void flip();
};

#endif
```

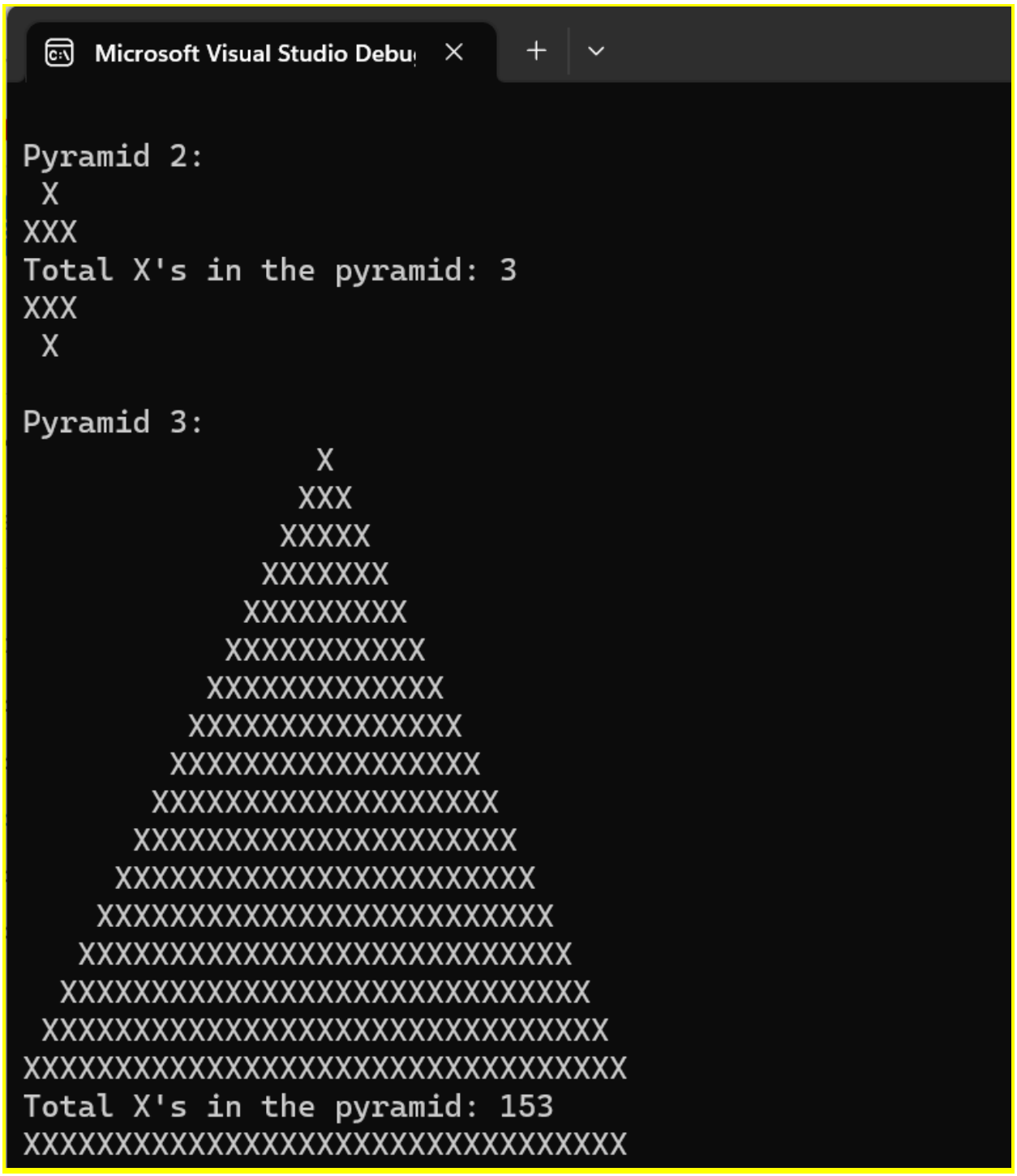
Pyramid.cpp

```cpp
//Naafiul Hossain
//SBU ID: 115107623
#include <iostream>
#include "pyramid.h"

Pyramid::Pyramid(int height) : height(height) {}

void Pyramid::create() {
    int num_x = 1;

    for (int i = 0; i < height; ++i) {
        for (int j = 0; j < height - i - 1; ++j)
            std::cout << " ";

        for (int k = 0; k < num_x; ++k)
            std::cout << "X";

        std::cout << "\n";
        num_x += 2;
    }

    std::cout << "Total X's in the pyramid: " << (height * (height + 1)) / 2 << "\n";
}

void Pyramid::flip() {
    int num_x = height * 2 - 1;

    for (int i = height - 1; i >= 0; --i) {
        for (int j = 0; j < height - i - 1; ++j)
            std::cout << " ";

        for (int k = 0; k < num_x; ++k)
            std::cout << "X";

        std::cout << "\n";
        num_x -= 2;
    }
}
```

Main.cpp

```cpp
//Naafiul Hossain
//SBU ID: 115107623
#include <iostream>
#include "pyramid.h"

int main() {
    Pyramid pyramid1(1);
    Pyramid pyramid2(2);
    Pyramid pyramid3(17);
    Pyramid pyramid4(20);
    Pyramid pyramid5(34);

    std::cout << "Pyramid 1:\n";
    pyramid1.create();
    pyramid1.flip();
    std::cout << "\n";

    std::cout << "Pyramid 2:\n";
    pyramid2.create();
    pyramid2.flip();
    std::cout << "\n";
```

Screenshot of the running program:



Problem 4

Lines.h

```
5
6   class Line {
7   private:
8       double slope;     // slope of the line (k)
9       double intercept; // y-intercept of the line (b)
10
11  public:
12      // Constructors
13      Line(double slope = 0.0, double intercept = 0.0);
14
15      // Print function
16      void print() const;
17
18      // Overloaded operator ==
19      bool operator==(const Line& other) const;
20
21      // Overloaded operator *
22      Line operator*(double multiplier) const;
23
24      // Overloaded operator +
25      Line operator+(double value) const;
26  };
```

Lines.cpp

```
1   #include "line.h"
2   #include <iostream>
3
4    // Constructor definition
5    Line::Line(double slope, double intercept) : slope(slope), intercept(intercept) {}
6
7    // Print function definition
8   void Line::print() const {
9       std::cout << "y = " << slope << "x + " << intercept << std::endl;
0   }
1
2    // Overloaded operator == definition
3   bool Line::operator==(const Line& other) const {
4       return (slope == other.slope) && (intercept == other.intercept);
5   }
6
7    // Overloaded operator * definition
8   Line Line::operator*(double multiplier) const {
9       return Line(slope * multiplier, intercept);
0   }
```

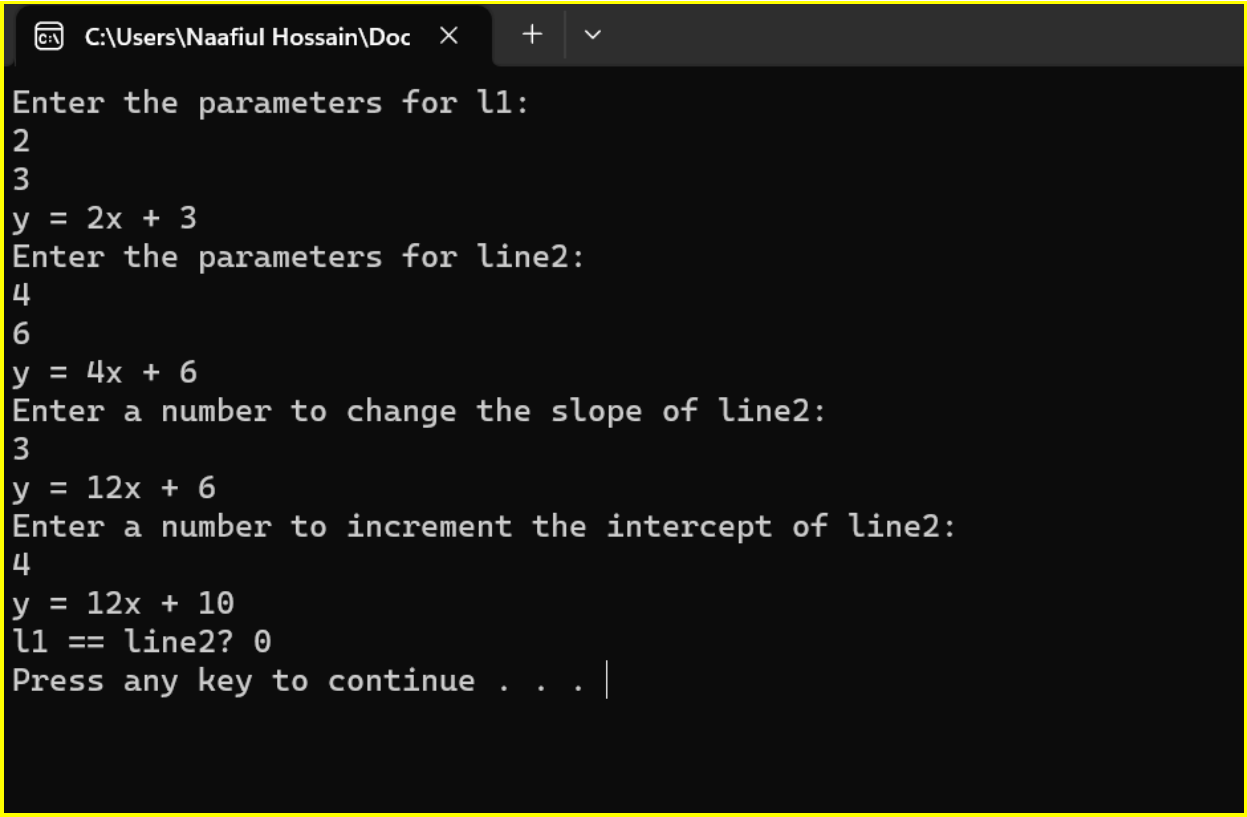
```cpp
// Overloaded operator * definition
Line Line::operator*(double multiplier) const {
    return Line(slope * multiplier, intercept);
}

// Overloaded operator + definition
Line Line::operator+(double value) const {
    return Line(slope, intercept + value);
}
```

Main.cpp

```cpp
//Naafiul Hossain
//SBU ID: 115107623
#include "line.h"
#include <iostream>

using namespace std;

int main() {
    double k, b, ck, cb;

    cout << "Enter the parameters for l1: " << endl;
    cin >> k >> b;
    Line l1(k, b);
    l1.print();

    cout << "Enter the parameters for line2: " << endl;
    cin >> k >> b;
    Line line2(k, b);
    line2.print();
```

```
    cout << "Enter a number to change the slope of line2: " << endl;
    cin >> ck;
    line2 = line2 * ck;
    line2.print();

    cout << "Enter a number to increment the intercept of line2: " << endl;
    cin >> cb;
    line2 = line2 + cb;
    line2.print();

    cout << "l1 == line2? " << (line2 == l1) << endl;

    system("pause");
    return 0;
}
```

Screenshot of the running program:



Problem 5

Games .h

```cpp
//Naafiul Hossain
//SBU ID: 115107623
#pragma once
#ifndef HEADER_H
#define HEADER_H

#include <iostream>
#include <cstdlib>   // For rand() and srand()
#include <ctime>     // For time()
#include <string>    // For string operations

// Function prototypes
void displayMenu();
int generateRandomNumber();
void addition();
void subtraction();
void multiplication();
void division();

#endif // HEADER_H
```

Games.cpp

```cpp
//Naafiul Hossain
//SBU ID: 115107623
#include "Game.h"

int generateRandomNumber() {
    // Seed the random number generator
    std::srand(static_cast<unsigned>(std::time(0)));

    // Generate a random number between 10 and 99
    return std::rand() % 90 + 10;
}

void displayMenu() {
    std::cout << "Input a number 1 - 5 to select a random problem or exit the game.\n";
    std::cout << "1 - Addition\n";
    std::cout << "2 - Subtraction\n";
    std::cout << "3 - Multiplication\n";
    std::cout << "4 - Division\n";
    std::cout << "5 - Exit\n";
}
```

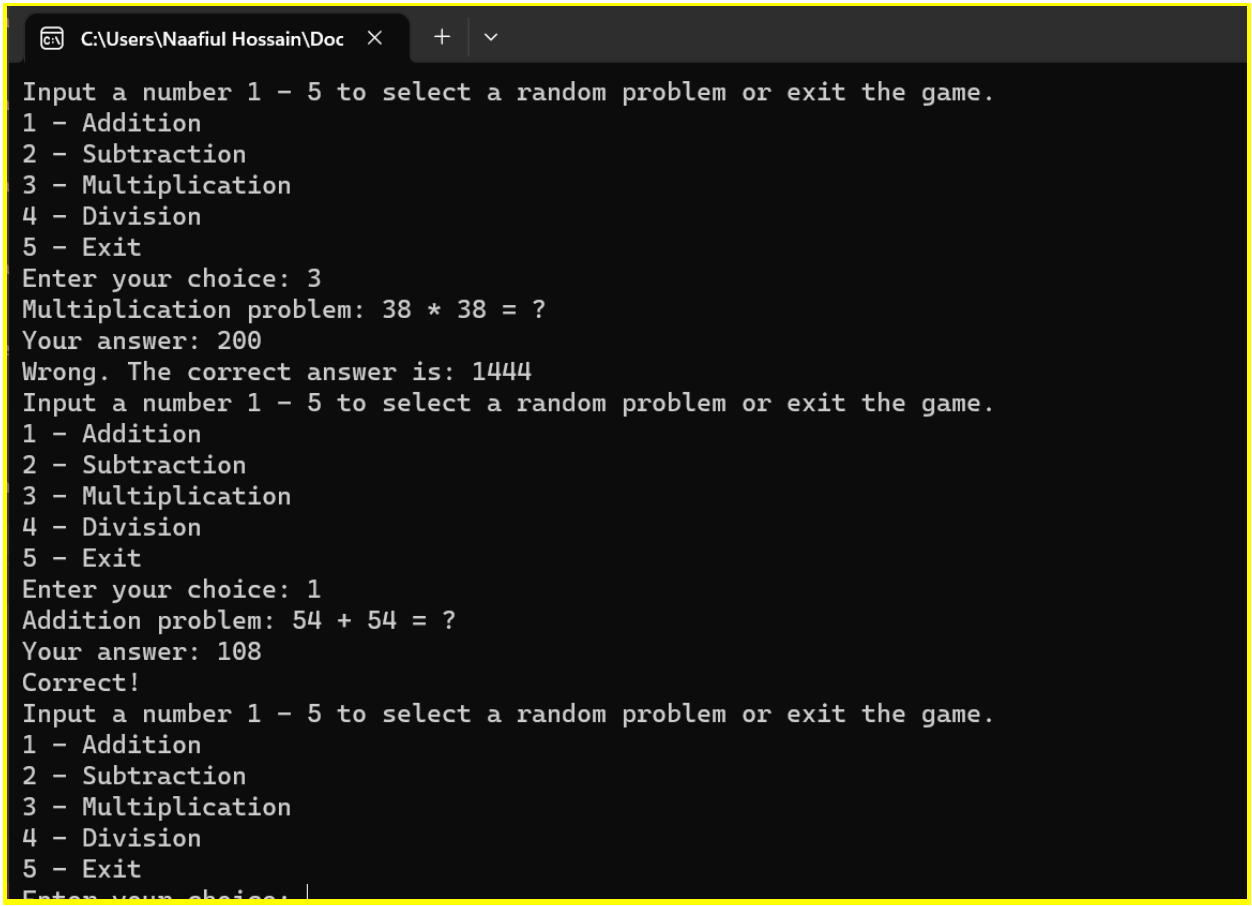
```cpp
void division() {
    int num1 = generateRandomNumber();
    int num2 = generateRandomNumber();

    // Ensure num2 is not 0 to avoid division by zero
    while (num2 == 0)
        num2 = generateRandomNumber();

    int result = num1 / num2;

    int userAnswer;
    std::cout << "Division problem: " << num1 << " / " << num2 << " = ?\n";
    std::cout << "Your answer: ";
    std::cin >> userAnswer;

    if (userAnswer == result)
        std::cout << "Correct!\n";
    else
        std::cout << "Wrong. The correct answer is: " << result << "\n";
}
```

Main.cpp

```cpp
//Naafiul Hossain
//SBU ID: 115107623
#include "Game.h"

int main() {
    srand(static_cast<unsigned>(time(0))); // Seed the random number generator

    for (;;) {
        displayMenu();

        int choice;
        std::cout << "Enter your choice: ";
        std::cin >> choice;

        switch (choice) {
        case 1:
            addition();
            break;
        case 2:
            subtraction();
            break;
        case 3:
            multiplication();
```

```
            break;
        case 2:
            subtraction();
            break;
        case 3:
            multiplication();
            break;
        case 4:
            division();
            break;
        case 5:
            std::cout << "Exiting the game.\n";
            return 0;
        default:
            std::cout << "Invalid choice. Please try again.\n";
        }
    }

    return 0;
}
```

Screenshot of the running program:



Problem 6 (Extra Credit)

Main.cpp

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

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

```cpp
bool isPalindrome(int num) {
    int originalNum = num;
    int reverseNum = 0;

    while (num > 0) {
        int digit = num % 10;
        reverseNum = reverseNum * 10 + digit;
        num /= 10;
    }

    return originalNum == reverseNum;
}
```

```cpp
int main() {
    int num;

    std::cout << "Enter an integer: ";
    std::cin >> num;

    if (isPalindrome(num)) {
        std::cout << num << " is a palindrome.\n";
    }
    else {
        std::cout << num << " is not a palindrome.\n";
    }

    return 0;
}
```

Running Solution:

```
Enter an integer: 1111
1111 is a palindrome.

C:\Users\Naafiul Hossain\Documents\Lab2\Lab1Task1\NHLab2Task6ExtraCredit\x64\Debug\NHLab2Task6ExtraCredit.exe (process 8
880) 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 . . .
```

```
Enter an integer: 1011
1011 is not a palindrome.

C:\Users\Naafiul Hossain\Documents\Lab2\Lab1Task1\NHLab2Task6ExtraCredit\x64\Debug\NHLa
8508) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debuggi
le when debugging stops.
Press any key to close this window . . .
```