Lab 5

Slope

Objective:

To write a program with:

- Proper formatting of program <u>with comments</u>
- Calculations using analytic geometry
- An <u>efficient</u> if-else construct (the order of the testing should be considered)

Assignment:

The slope m of the line between the two points (x_1, y_2) and (x_2, y_2) is given by the formula:

$$\mathbf{m = rise/run = (y_2 - y_1)/(x_2 - x_1)}$$

Write a program that will ask the user for the coordinates of two points and do the following:

- Determine if the two points form a line.
- Detect if the line is horizontal and alert the user.
- Detect if the line is vertical and alert the user.
- Calculate the slope of the line segment connecting them, if not 0 or undefined.

Sample Run:

```
Enter the coordinates of point 1 (x, y): 4 9
Enter the coordinates of point 2 (x, y): 4 9
Alert! These points do not form a line.
```

Sample Run:

```
Enter the coordinates of point 1 (x, y): 3 5
Enter the coordinates of point 2 (x, y): 5 8
The slope of the line is 1.50
```

Sample Run:

```
Enter the coordinates of point 1 (x, y): -3 7
Enter the coordinates of point 2 (x, y): 5 7
Alert! The line is horizontal.
```

Sample Run:

```
Enter the coordinates of point 1 (x, y): 6 -8
Enter the coordinates of point 2 (x, y): 6 12
Alert! The line is vertical.
```

Next, modify the program to print the equation of the line in slope-intercept form. Use the appropriate “y=” or “x=” for horizontal and vertical lines, respectively.

Sample Run:

```
Enter the coordinates of point 1 (x, y): 3 5
Enter the coordinates of point 2 (x, y): 5 8
The slope of the line is 1.50
The slope-intercept form is y = 1.50x + .50
```

Sample Run:

```
Enter the coordinates of point 1 (x, y): 3 5
Enter the coordinates of point 2 (x, y): 3 8
Alert! The line is horizontal.
The equation of the line is x = 3.00
```

Hint:

To format a variable to two decimal places, include the header file iomanip and use these cout class manipulators:

```
cout << setiosflags(ios_base::fixed)              // do not use E notation
     << setiosflags(ios_base::showpoint)          // always show decimal point
     << setprecision(2);                          // rounded to 2 decimal places
```

Placing the above code at the beginning of your program, inside `int main()`, forces the formatting to occur throughout the running of your program.

You may find the If-Else and Variables Input and Output Formatting reference documents useful.