PROG2070 – Quality Assurance: Assignment 2

[Maximum points: 50]

Due Date: Day of your Class, Week of Feb. 8 – Feb. 12

This assignment should be done **individually**. Do your own work and **do not share** your work with others. Sharing work is an academic offense and is subject to penalty. Be aware that source code and documents are automatically checked by eConestoga against every other student's work in the course. Academic offenses will be reported to the College Registrar.

This assignment will be demonstrated in class. Any assignment not demonstrated in class will receive a 20% penalty.

You are to create a C# console application that first presents the user with the following menu [1 Mark]:

1. **Enter triangle dimensions**
2. **Exit**

If any incorrect input is given, the menu is shown again [1 Mark]. Your program should never crash.

The exit option quits the program, and is the only way to exit the program (other than closing the window) [1 Mark].

If the user selects the first option, the program asks the user to enter three integers. The program will use these three numbers and report back to the user if the numbers:

A. Form a triangle or not [2 Marks]
B. If the numbers do form a triangle, the program reports if that triangle is scalene, isosceles, or equilateral. [6 Marks]

You must create a TriangleFinder class. This class contains a non-static method called Analyze which takes three integers as an input, and returns a string as an output [5 Marks]. The Triangle class must be public, and should contain a default Constructor [1 Mark].

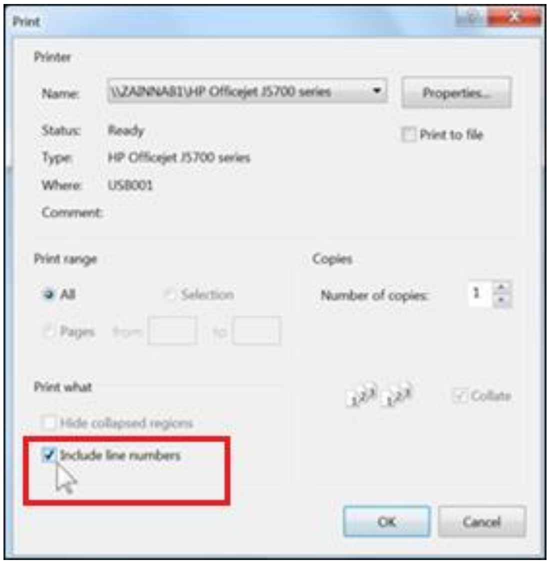
Write eight unit tests for the Analyze method [8 marks]. Each unit test should test some different condition or boundary. Provide a briefly (one or two sentences) explaining why you selected each input for your unit cases [8 Marks]. Provide a screenshot of the Unit cases being run (can take a screenshot of the NUnit GUI after a run) [1 Mark]. All of your unit tests should pass.

Your test cases should be in their own project, but in the same solution as your console program, as you did in Assignment 1. Remember the naming conventions for the Test Project, the Test Class, and the Test Methods.

Draw Control Flow Graphs for both your Program.cs and your Analyze method. The CFGs should be drawn in an appropriate modeling software (such as Microsoft Visio). Label all of your nodes with the line numbers that match your source code [10 Marks].

Finally, compute the Cyclomatic Complexity of your two CFGs, showing your work [4 Marks]. Provide a brief description of what the Cyclomatic Complexity number of each CFG tells you about your solution [2 Marks].

When printing your source code, please include the line numbers. This can be done in Visual Studio by checking the "Include line numbers" check box on the print prompt window.



Printer

Name: \\ZAINNAB1\HP Officejet J5700 series

Status: Ready

Type: HP Officejet J5700 series

Where: USB001

Comment:

□ Print to file

Print range

● All ○ Selection

○ Pages from to

Copies

Number of copies: 1

☑ Collate

Print what

□ Hide collapsed regions

☑ Include line numbers

The format for submitting the assignment is as follows:

1. **Demonstrate your program in class.**
2. **Printouts Handed in Class**:
    a. Assignment Cover sheet properly filled (found on eConestoga)
    b. Copy of Program source code
    c. Copy of TriangleFinder class source code
    d. Copy of Unit Test class source code
    e. Print out of your two CFGs
    f. Print out of Doc showing the results of your unit tests being run, with an explanation to why you chose each case. This doc should also include your two CFGs, and a Cyclomatic Complexity calculation for each. Finally, a brief discussion of the two CC values should be included
3. **eConestoga Submission**: A single compressed (.zip format) archive file containing the project folder of your source code (so I can run it) and the doc file with the screenshots and explanations.

Marks for assignment 1 will be given as follows:

Marks for proper Console application:	11
Marks for proper Circle class:	6
Marks for Unit Test class:	17
Two Control Flow Graphs:	10
Cyclomatic Complexity:	6
Non-demonstration Penalty:	-10
Deduction for coding error:	-0.25 per
Total:	50