Assignment 2: Triangles [0%]

This assignment has zero weighting (it doesn't contribute to your unit mark)
You should however still submit it as you will get valuable feedback on the quality of your code

Overview

In this exercise you will write a Triangle class to represent triangles.
This class will have a single constructor method that takes three integer parameters
(the length of each side of the triangle)

The class will be "self-aware": it will know what kind of triangle it is
(Equilateral, Isosceles, Right-angle etc.)
You won't tell the triangle what type it is, the triangle will work it out for itself
(based on the length of the sides passed in).

You will be given a template/skeleton project to help get you started.
You will need to complete some of the existing methods in the *Triangle* class,
as well as adding some methods of your own.
TriangleType is an enumeration that represents the range of triangle types that you will need.

How to Proceed

Created a multi-branch IF statement that will allow the triangle to decide what kind it is.
Note: The order of the branches of the IF statements will be very important
Check for specific/special cases at the top, with more general cases at the bottom
(you'll soon understand why !)

The *TriangleTester* class contains a variety of test cases to help you develop your class.
(run `make` to compile your class or `make test` to run the test cases against it)

Focus on a particular type of triangle and add code to your class to identify that type.
Make sure your code passes all of the tests for that type before moving on to the next.

The final group of tests (to do with overflow) are more difficult, so leave those until the end.
These may require you to refactor your code a fair bit !

Submission

This assignment is not assessed, however it will be good practice to submit it onto blackboard
(go to "Assessment, submission and feedback" and then "Other Formative and Draft Submissions").

It is **essential** that you ensure your code compiles and runs before you submit it
(otherwise we will not be able to run it to mark it !)
Just Submit your *Triangle* class (nothing else should be needed)
Scripts will be used to automatically test your code to make sure it operates correctly.
Don't change the name of the class or the signatures of any of the methods that already exist
(or we won't be able to test your code !).