Initial Triangle.py HW02a

The objective of this assignment is to (a) develop a set of tests for an existing triangle classification program, (b) use those tests to find and fix defects in that program, and (c) report on your testing results for the Triangle problem. You will start with an existing implementation of the classify triangle program that will be given to you and a starter test program that tests the classify triangle program, but those tests are not complete. In order to determine if the program is correctly implemented, you will need to update the set of test cases in the test program. You will need to update the test program until you feel that your tests adequately test all of the conditions. Then you should run the complete set of tests against the original triangle program to see how correct the triangle program is. Report on those results in a formal test report.

Sarah Wiessler

Summary

The results showed that all tests failed due to either a TypeError when the input was not an integer or “InvalidInput” for all other test cases. I found this peculiar because on initial observation of the Triangle.py file, it did not seem like every test should have resulted in InvalidInput. The inputs given appear as if they should have passed the initial input checking. Additionally, based on the program requirements I maintain that “InvalidInput” should not be returned. In case of invalid input, “NotATriangle” should be returned based off the specifications.

I pledge my honor that I have abided by the Stevens Honor System.

Detailed Results

Constraints

Constraints applied by the program stated that input must be and integers need to be within 0-200 range. Additionally, the sum of any two sides must be less than the third.

Data Inputs

Integers, strings, list, float

Results

Test ID	Input	Expected Results	Actual Result	Pass or Fail
testInvalidInputA	“zoinks”,4,5	NotATriangle	TypeError: '>' not supported between instances of 'str' and 'int'	Fail
testInvalidInputD	[20],4,4	NotATriangle	TypeError: '>' not supported between	Fail

			instances of 'list' and 'int	
testEquilateralTriangleA	4,4,4	Equilateral	InvalidInput	Fail
testInvalidInputB	-1,4,4	NotATriangle	InvalidInput	Fail
testInvalidInputC	0,4,4	NotATriangle	InvalidInput	Fail
testIsocelesTriangleA	10.5,10.5,20	Isoceles	InvalidInput	Fail
testIsocelesTriangleB	11,11,20	Isoceles	InvalidInput	Fail
testNonviableSideA	3,4,8	NotATriangle	InvalidInput	Fail
testRightTriangleA	3,4,5	Right	InvalidInput	Fail
testRightTriangleB	5,3,4	Right	InvalidInput	Fail
testScaleneTriangleA	10,15,20	Scalene	InvalidInput	Fail