**HW 02a – Testing a legacy program and reporting on testing results.**

Name: Saurabh Agrawal

1. Assignment Description:

The objective of this assignments is for you to (a) develop a set for an existing triangle classification program, (b) us those tests to find and fix defects in that program, and (c) report on your testing results for the Triangle problem.

1. Author: Saurabh Agrawal
2. Summary:

The test cases below depicts that the initial code has lots of bugs with the logic and flow of the code as shown by TestRun1.

After the changes were made to the code, the TestRun2 gave 6 passed and 3 failed cases of assertion errors. They were corrected and TestRun3 showcases that ever test has passes.

1. Honor pledge: I pledge my honour that I have abided by the Stevens Honor System
2. Detailed Results:

Test Report 1:

This test report contains the result of running TestTriangle on the original (buggy) triangle.py

Running unit tests

F.FFFFFFFFF

======================================================================

FAIL: testEquilateralTriangleB (\_\_main\_\_.TestTriangles)

----------------------------------------------------------------------

Traceback (most recent call last):

File "c:\Users\saura\OneDrive\Desktop\SSE\567\New folder\TestTriangle.py", line 32, in testEquilateralTriangleB

self.assertEqual(classifyTriangle(2,3,3),'Equilateral')

AssertionError: 'InvalidInput' != 'Equilateral'

- InvalidInput

+ Equilateral

======================================================================

FAIL: testIsocelesTriangleA (\_\_main\_\_.TestTriangles)

----------------------------------------------------------------------

Traceback (most recent call last):

File "c:\Users\saura\OneDrive\Desktop\SSE\567\New folder\TestTriangle.py", line 35, in testIsocelesTriangleA

self.assertEqual(classifyTriangle(3, 1, 2),'Isoceles')

AssertionError: 'InvalidInput' != 'Isoceles'

- InvalidInput

+ Isoceles

======================================================================

FAIL: testIsocelesTriangleC (\_\_main\_\_.TestTriangles)

----------------------------------------------------------------------

Traceback (most recent call last):

File "c:\Users\saura\OneDrive\Desktop\SSE\567\New folder\TestTriangle.py", line 39, in testIsocelesTriangleC

self.assertEqual(classifyTriangle(2, 1, 6),'Isoceles')

AssertionError: 'InvalidInput' != 'Isoceles'

- InvalidInput

+ Isoceles

======================================================================

FAIL: testIsocelesTriangleD (\_\_main\_\_.TestTriangles)

----------------------------------------------------------------------

Traceback (most recent call last):

File "c:\Users\saura\OneDrive\Desktop\SSE\567\New folder\TestTriangle.py", line 42, in testIsocelesTriangleD

self.assertEqual(classifyTriangle(2, 6, 4),'Isoceles')

AssertionError: 'InvalidInput' != 'Isoceles'

- InvalidInput

+ Isoceles

======================================================================

FAIL: testNotATriangleA (\_\_main\_\_.TestTriangles)

----------------------------------------------------------------------

Traceback (most recent call last):

File "c:\Users\saura\OneDrive\Desktop\SSE\567\New folder\TestTriangle.py", line 52, in testNotATriangleA

self.assertEqual(classifyTriangle(5, 1, 5),'NotATriangle')

AssertionError: 'InvalidInput' != 'NotATriangle'

- InvalidInput

+ NotATriangle

======================================================================

FAIL: testNotATriangleC (\_\_main\_\_.TestTriangles)

----------------------------------------------------------------------

Traceback (most recent call last):

File "c:\Users\saura\OneDrive\Desktop\SSE\567\New folder\TestTriangle.py", line 55, in testNotATriangleC

self.assertEqual(classifyTriangle(1, -1, 5),'NotATriangle')

AssertionError: 'InvalidInput' != 'NotATriangle'

- InvalidInput

+ NotATriangle

======================================================================

FAIL: testRightTriangleA (\_\_main\_\_.TestTriangles)

----------------------------------------------------------------------

Traceback (most recent call last):

File "c:\Users\saura\OneDrive\Desktop\SSE\567\New folder\TestTriangle.py", line 21, in testRightTriangleA

self.assertEqual(classifyTriangle(31,4,5),'Right')

AssertionError: 'InvalidInput' != 'Right'

- InvalidInput

+ Right

======================================================================

FAIL: testRightTriangleB (\_\_main\_\_.TestTriangles)

----------------------------------------------------------------------

Traceback (most recent call last):

File "c:\Users\saura\OneDrive\Desktop\SSE\567\New folder\TestTriangle.py", line 24, in testRightTriangleB

self.assertEqual(classifyTriangle(5,73,4),'Right')

AssertionError: 'InvalidInput' != 'Right'

- InvalidInput

+ Right

======================================================================

FAIL: testRightTriangleD (\_\_main\_\_.TestTriangles)

----------------------------------------------------------------------

Traceback (most recent call last):

File "c:\Users\saura\OneDrive\Desktop\SSE\567\New folder\TestTriangle.py", line 28, in testRightTriangleD

self.assertEqual(classifyTriangle(8, 9, 10), 'Right')

AssertionError: 'InvalidInput' != 'Right'

- InvalidInput

+ Right

PS C:\Users\saura> & "C:/Program Files/Python310/python.exe" "c:/Users/saura/OneDrive/Desktop/SSE/567/New folder/TestTriangle.py"

Running unit tests

F.FFFFFFFFF

======================================================================

FAIL: testEquilateralTriangleB (\_\_main\_\_.TestTriangles)

----------------------------------------------------------------------

Traceback (most recent call last):

File "c:\Users\saura\OneDrive\Desktop\SSE\567\New folder\TestTriangle.py", line 32, in testEquilateralTriangleB

self.assertEqual(classifyTriangle(2,3,3),'Equilateral')

AssertionError: 'InvalidInput' != 'Equilateral'

- InvalidInput

+ Equilateral

======================================================================

FAIL: testIsocelesTriangleA (\_\_main\_\_.TestTriangles)

----------------------------------------------------------------------

Traceback (most recent call last):

File "c:\Users\saura\OneDrive\Desktop\SSE\567\New folder\TestTriangle.py", line 35, in testIsocelesTriangleA

self.assertEqual(classifyTriangle(3, 1, 2),'Isoceles')

AssertionError: 'InvalidInput' != 'Isoceles'

- InvalidInput

+ Isoceles

======================================================================

FAIL: testIsocelesTriangleC (\_\_main\_\_.TestTriangles)

----------------------------------------------------------------------

Traceback (most recent call last):

File "c:\Users\saura\OneDrive\Desktop\SSE\567\New folder\TestTriangle.py", line 39, in testIsocelesTriangleC

self.assertEqual(classifyTriangle(2, 1, 6),'Isoceles')

AssertionError: 'InvalidInput' != 'Isoceles'

- InvalidInput

+ Isoceles

======================================================================

FAIL: testIsocelesTriangleD (\_\_main\_\_.TestTriangles)

----------------------------------------------------------------------

Traceback (most recent call last):

File "c:\Users\saura\OneDrive\Desktop\SSE\567\New folder\TestTriangle.py", line 42, in testIsocelesTriangleD

self.assertEqual(classifyTriangle(2, 6, 4),'Isoceles')

AssertionError: 'InvalidInput' != 'Isoceles'

- InvalidInput

+ Isoceles

======================================================================

FAIL: testNotATriangleA (\_\_main\_\_.TestTriangles)

----------------------------------------------------------------------

Traceback (most recent call last):

File "c:\Users\saura\OneDrive\Desktop\SSE\567\New folder\TestTriangle.py", line 52, in testNotATriangleA

self.assertEqual(classifyTriangle(5, 1, 5),'NotATriangle')

AssertionError: 'InvalidInput' != 'NotATriangle'

- InvalidInput

+ NotATriangle

======================================================================

FAIL: testNotATriangleC (\_\_main\_\_.TestTriangles)

----------------------------------------------------------------------

Traceback (most recent call last):

File "c:\Users\saura\OneDrive\Desktop\SSE\567\New folder\TestTriangle.py", line 55, in testNotATriangleC

self.assertEqual(classifyTriangle(1, -1, 5),'NotATriangle')

AssertionError: 'InvalidInput' != 'NotATriangle'

- InvalidInput

+ NotATriangle

======================================================================

FAIL: testRightTriangleA (\_\_main\_\_.TestTriangles)

----------------------------------------------------------------------

Traceback (most recent call last):

File "c:\Users\saura\OneDrive\Desktop\SSE\567\New folder\TestTriangle.py", line 21, in testRightTriangleA

self.assertEqual(classifyTriangle(1,4,5),'Right')

AssertionError: 'InvalidInput' != 'Right'

- InvalidInput

+ Right

======================================================================

FAIL: testRightTriangleB (\_\_main\_\_.TestTriangles)

----------------------------------------------------------------------

Traceback (most recent call last):

File "c:\Users\saura\OneDrive\Desktop\SSE\567\New folder\TestTriangle.py", line 24, in testRightTriangleB

self.assertEqual(classifyTriangle(5,7,4),'Right')

AssertionError: 'InvalidInput' != 'Right'

- InvalidInput

+ Right

======================================================================

FAIL: testRightTriangleD (\_\_main\_\_.TestTriangles)

----------------------------------------------------------------------

Traceback (most recent call last):

File "c:\Users\saura\OneDrive\Desktop\SSE\567\New folder\TestTriangle.py", line 28, in testRightTriangleD

self.assertEqual(classifyTriangle(8, 9, 10), 'Right')

AssertionError: 'InvalidInput' != 'Right'

- InvalidInput

+ Right

======================================================================

FAIL: testScaleneTriangleA (\_\_main\_\_.TestTriangles)

----------------------------------------------------------------------

Traceback (most recent call last):

File "c:\Users\saura\OneDrive\Desktop\SSE\567\New folder\TestTriangle.py", line 45, in testScaleneTriangleA

self.assertEqual(classifyTriangle(15, 15, 12),'Scalene')

AssertionError: 'InvalidInput' != 'Scalene'

- InvalidInput

+ Scalene

----------------------------------------------------------------------

Ran 11 tests in 0.003s

FAILED (failures=10)

**Test Report 2:**

This test report contains the result of running TestTriangle on the new updated (fixed) triangle.py code.

PS C:\Users\saura> & "C:/Program Files/Python310/python.exe" "c:/Users/saura/OneDrive/Desktop/SSE/567/New folder/TestTriangle.py"

Running unit tests

...........

----------------------------------------------------------------------

Ran 11 tests in 0.002s

OK

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test ID | Input | Expected output | Actual Result | Pass or Fail |
| testRightTriangleA | (3,4,5) | Right | Right | Pass |
| testRightTriangleB | (5,3,4) | Right | Right | Pass |
| testRightTriangleC | (13,12,5) | Right | Right | Pass |
| testEquilateralTriangleA | (1,1,1) | Equilateral | Equilateral | Pass |
| testIsocelesTriangleD | (6,6,4) | Isosceles | Isosceles | Pass |
| testScaleneTriangleA | (10,15,12) | Scalene | Scalene | Pass |
| testInvalidInputB | (201,201,201) | InvalidInput | InvalidInput | Pass |
| testInvalidInputC | (200,0,0) | InvalidInput | InvalidInput | Pass |
| testNotATriangleA | (5,1,1) | NotATriangle | NotATriangle | Pass |
| testNotATriangleB | (1,5,1) | NotATriangle | NotATriangle | Pass |
| testNotATriangleC | (1,1,5) | NotATriangle | NotATriangle | Pass |

Final Test Summary:

|  |  |  |
| --- | --- | --- |
|  | TestRun1 | TestRun2 |
| Tests Planned | 11 | 11 |
| Tests Executed | 11 | 11 |
| Tests Passed | 0 | 11 |
| Defects Found | 11 | 0 |
| Defects Fixed | 0 | 0 |

The test cases below depicts that the initial code has a lot of bugs with the logic and flow of the code as shown by TestRun1.