1. Include a written test report in your assignment summary with the results of running your test set against the initial buggy implementation of classifyTriangle in the original Triangle.py using  the following format:

Summary :- This test cases report is for buggy Triangle.py in which the classifyTriangle has some bugs. In this function there is some condition did not work properly.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test ID** | **Input** | **Actual Result** | **Expected Result** | **Pass** **or** **Fail** |
| testRightTriangleA | (3,4,5) | InvalidInput | Right | Fail |
| testRightTriangleB | (5,3,4) | InvalidInput | Right | Fail |
| testEquilateralTrianglesA | (1,1,1) | InvalidInput | Equilateral | Fail |
| testEquilateralTrianglesB | (5,5,5) | InvalidInput | Equilateral | Fail |
| testScaleneTrianglesA | (8,11,13) | InvalidInput | Scalene | Fail |
| testScaleneTrianglesB | (15,34,32) | InvalidInput | Scalene | Fail |
| testIsosceleneTriangles | (9,9,3) | InvalidInput | Isoscelene | Fail |
| testIsNotATriangleA | (1,2,10) | InvalidInput | NotATriangle | Fail |
| testIsNotATriangleB | (5,6,20) | InvalidInput | NotATriangle | Fail |
| testInvalidInputTriangleA | (-5,-6,20) | InvalidInput | InvalidInput | Pass |
| testInvalidInputTriangleB | (5,6,201) | InvalidInput | InvalidInput | Pass |

Output:-

/Users/neelsavani/PycharmProjects/pythonProject/venv/bin/python /Applications/PyCharm CE.app/Contents/plugins/python-ce/helpers/pycharm/\_jb\_unittest\_runner.py --target TestTriangle.TestTriangles

Testing started at 4:35 am ...

Ran 11 tests in 0.007s

FAILED (failures=9)

Launching unittests with arguments python -m unittest TestTriangle.TestTriangles in /Users/neelsavani/PycharmProjects/pythonProject

1,1,1 should be equilateral

Equilateral != InvalidInput

Expected :InvalidInput

Actual :Equilateral

<Click to see difference>

Traceback (most recent call last):

File "/Users/neelsavani/PycharmProjects/pythonProject/TestTriangle.py", line 15, in testEquilateralTrianglesA

self.assertEqual(classifyTriangle(1, 1, 1), 'Equilateral', '1,1,1 should be equilateral')

AssertionError: 'InvalidInput' != 'Equilateral'

- InvalidInput

+ Equilateral

: 1,1,1 should be equilateral

5,5,5 should be equilateral

Equilateral != InvalidInput

Expected :InvalidInput

Actual :Equilateral

<Click to see difference>

Traceback (most recent call last):

File "/Users/neelsavani/PycharmProjects/pythonProject/TestTriangle.py", line 18, in testEquilateralTrianglesB

self.assertEqual(classifyTriangle(5, 5, 5), 'Equilateral', '5,5,5 should be equilateral')

AssertionError: 'InvalidInput' != 'Equilateral'

- InvalidInput

+ Equilateral

: 5,5,5 should be equilateral

1,2,10 should not be triangle

NotATriangle != InvalidInput

Expected :InvalidInput

Actual :NotATriangle

<Click to see difference>

Traceback (most recent call last):

File "/Users/neelsavani/PycharmProjects/pythonProject/TestTriangle.py", line 30, in testIsNotTrianglesA

self.assertEqual(classifyTriangle(1, 2, 10), 'NotATriangle', '1,2,10 should not be triangle')

AssertionError: 'InvalidInput' != 'NotATriangle'

- InvalidInput

+ NotATriangle

: 1,2,10 should not be triangle

5,6,20 should not be triangle

NotATriangle != InvalidInput

Expected :InvalidInput

Actual :NotATriangle

<Click to see difference>

Traceback (most recent call last):

File "/Users/neelsavani/PycharmProjects/pythonProject/TestTriangle.py", line 33, in testIsNotTrianglesB

self.assertEqual(classifyTriangle(5, 6, 20), 'NotATriangle', '5,6,20 should not be triangle')

AssertionError: 'InvalidInput' != 'NotATriangle'

- InvalidInput

+ NotATriangle

: 5,6,20 should not be triangle

9,9,3 should be Isosceles

Isosceles != InvalidInput

Expected :InvalidInput

Actual :Isosceles

<Click to see difference>

Traceback (most recent call last):

File "/Users/neelsavani/PycharmProjects/pythonProject/TestTriangle.py", line 27, in testIsoscelesTriangles

self.assertEqual(classifyTriangle(9, 9, 3), 'Isosceles', '9,9,3 should be Isosceles')

AssertionError: 'InvalidInput' != 'Isosceles'

- InvalidInput

+ Isosceles

: 9,9,3 should be Isosceles

3,4,5 is a Right triangle

Right != InvalidInput

Expected :InvalidInput

Actual :Right

<Click to see difference>

Traceback (most recent call last):

File "/Users/neelsavani/PycharmProjects/pythonProject/TestTriangle.py", line 9, in testRightTriangleA

self.assertEqual(classifyTriangle(3, 4, 5), 'Right', '3,4,5 is a Right triangle')

AssertionError: 'InvalidInput' != 'Right'

- InvalidInput

+ Right

: 3,4,5 is a Right triangle

5,3,4 is a Right triangle

Right != InvalidInput

Expected :InvalidInput

Actual :Right

<Click to see difference>

Traceback (most recent call last):

File "/Users/neelsavani/PycharmProjects/pythonProject/TestTriangle.py", line 12, in testRightTriangleB

self.assertEqual(classifyTriangle(5, 3, 4), 'Right', '5,3,4 is a Right triangle')

AssertionError: 'InvalidInput' != 'Right'

- InvalidInput

+ Right

: 5,3,4 is a Right triangle

8,11,13 should be Scalene

Scalene != InvalidInput

Expected :InvalidInput

Actual :Scalene

<Click to see difference>

Traceback (most recent call last):

File "/Users/neelsavani/PycharmProjects/pythonProject/TestTriangle.py", line 21, in testScaleneTrianglesA

self.assertEqual(classifyTriangle(8, 11, 13), 'Scalene', '8,11,13 should be Scalene')

AssertionError: 'InvalidInput' != 'Scalene'

- InvalidInput

+ Scalene

: 8,11,13 should be Scalene

15,34,32 should be Scalene

Scalene != InvalidInput

Expected :InvalidInput

Actual :Scalene

<Click to see difference>

Traceback (most recent call last):

File "/Users/neelsavani/PycharmProjects/pythonProject/TestTriangle.py", line 24, in testScaleneTrianglesB

self.assertEqual(classifyTriangle(15, 34, 32), 'Scalene', '15,34,32 should be Scalene')

AssertionError: 'InvalidInput' != 'Scalene'

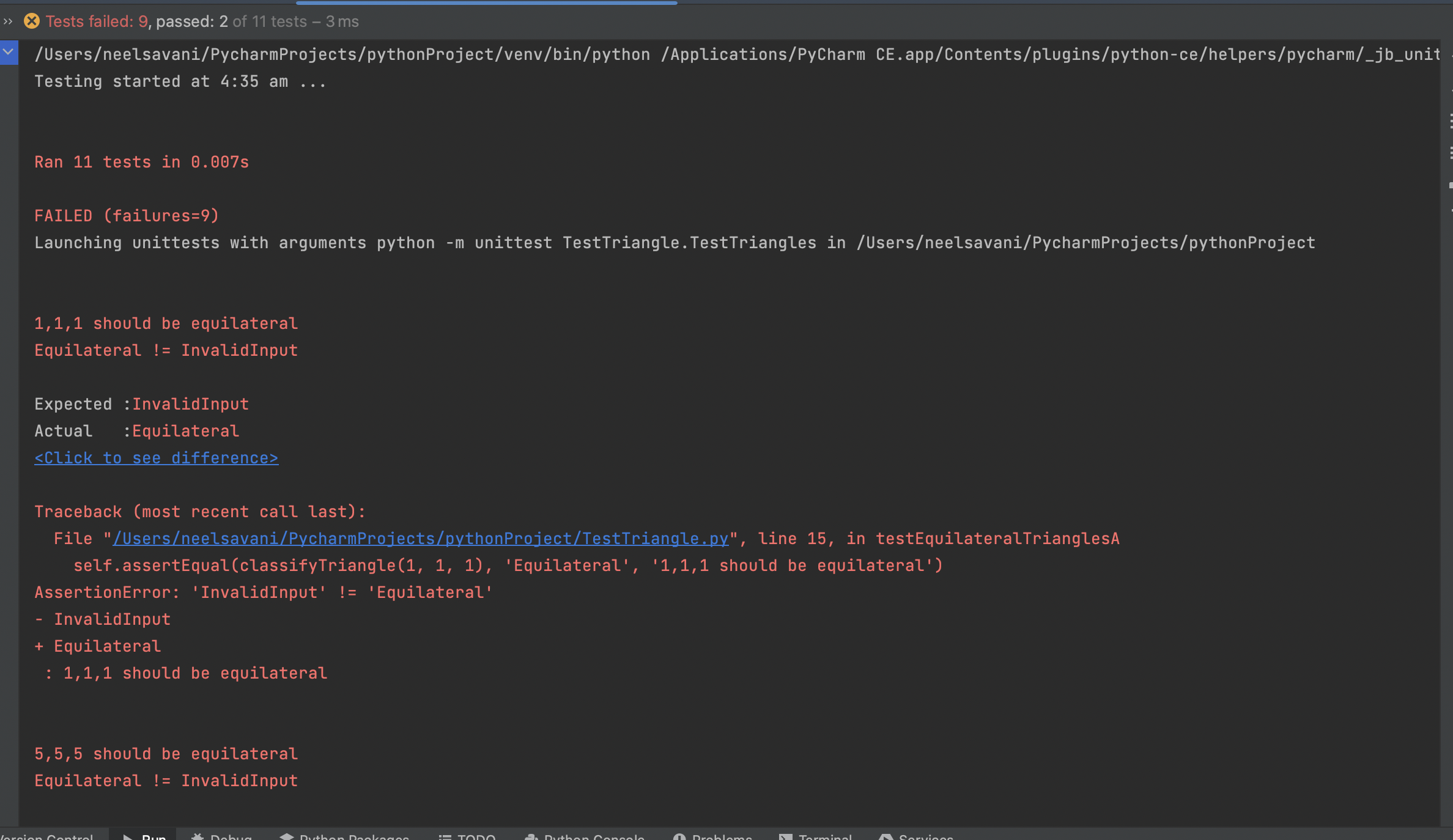
- InvalidInput

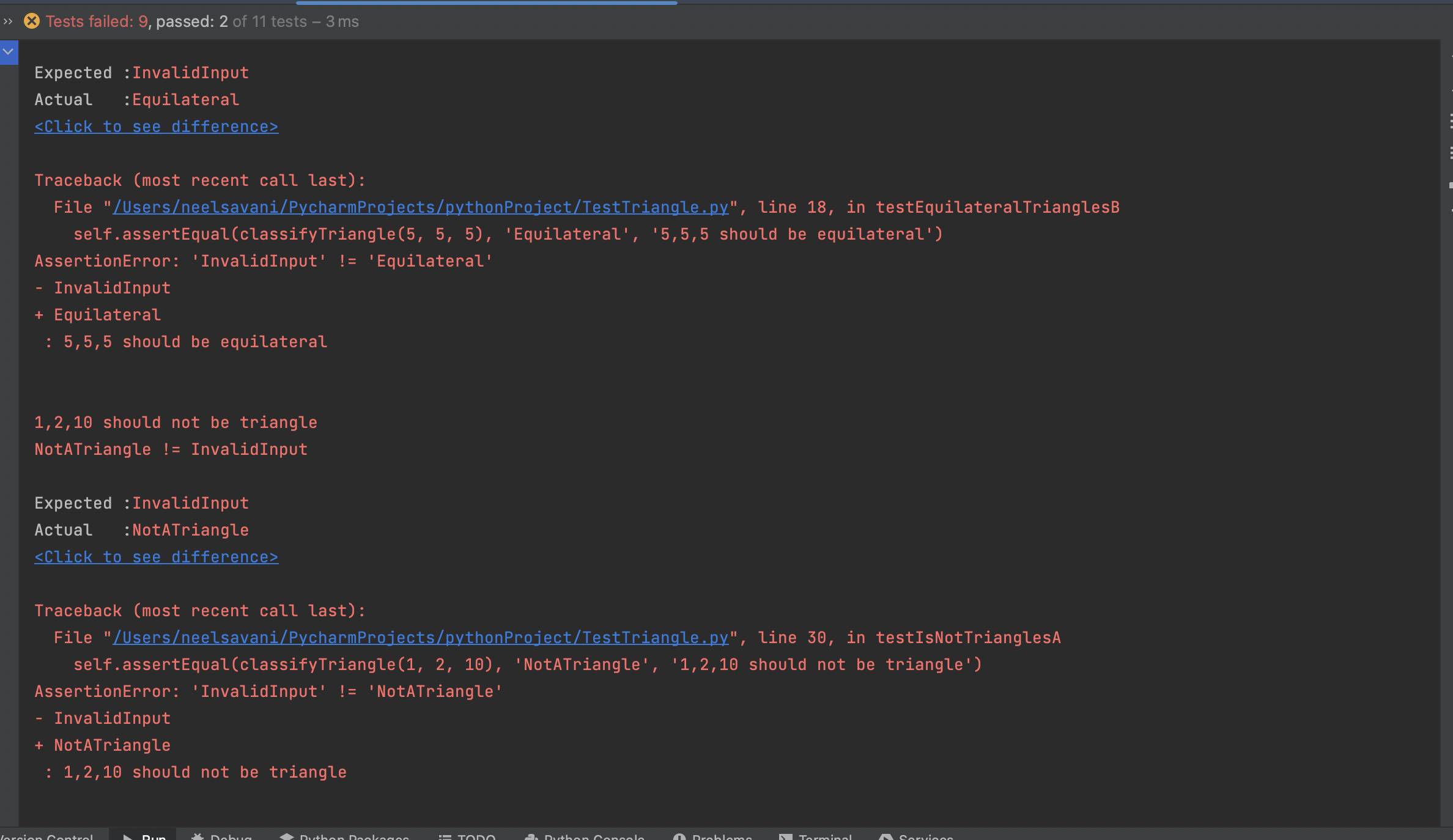
+ Scalene

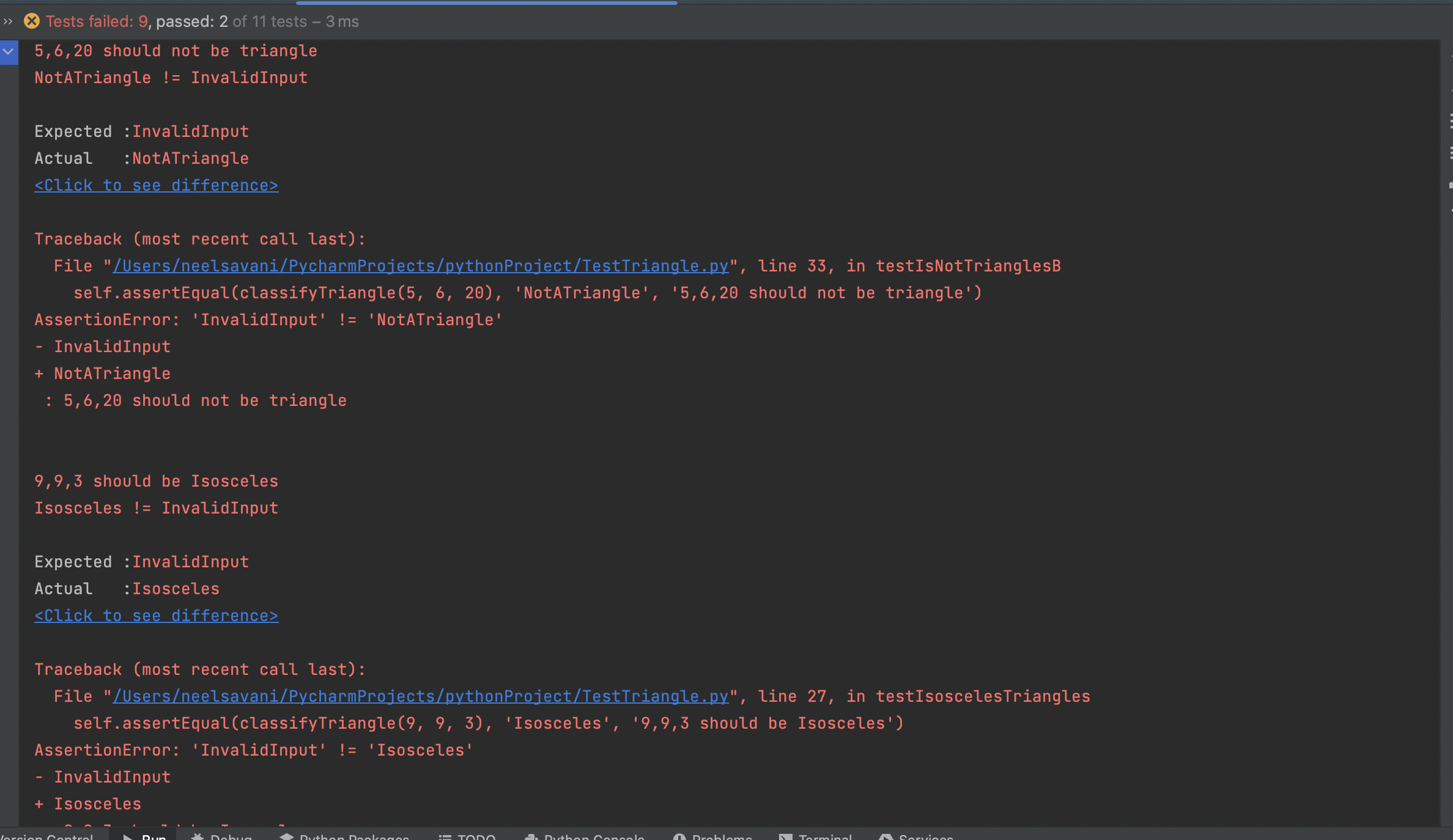
: 15,34,32 should be Scalene

Process finished with exit code 1

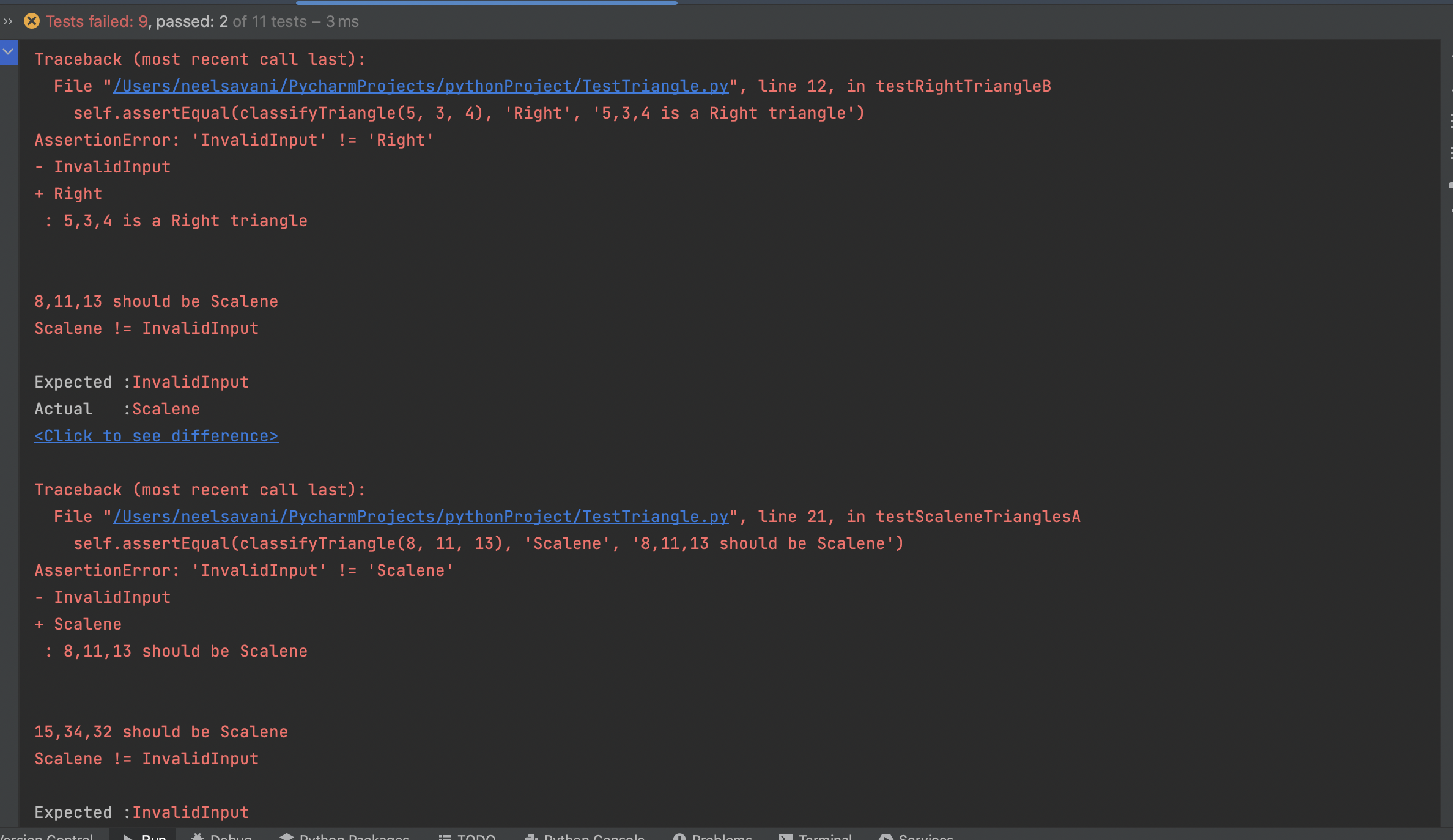
These are screenshots of output of buggy Triangle.py file

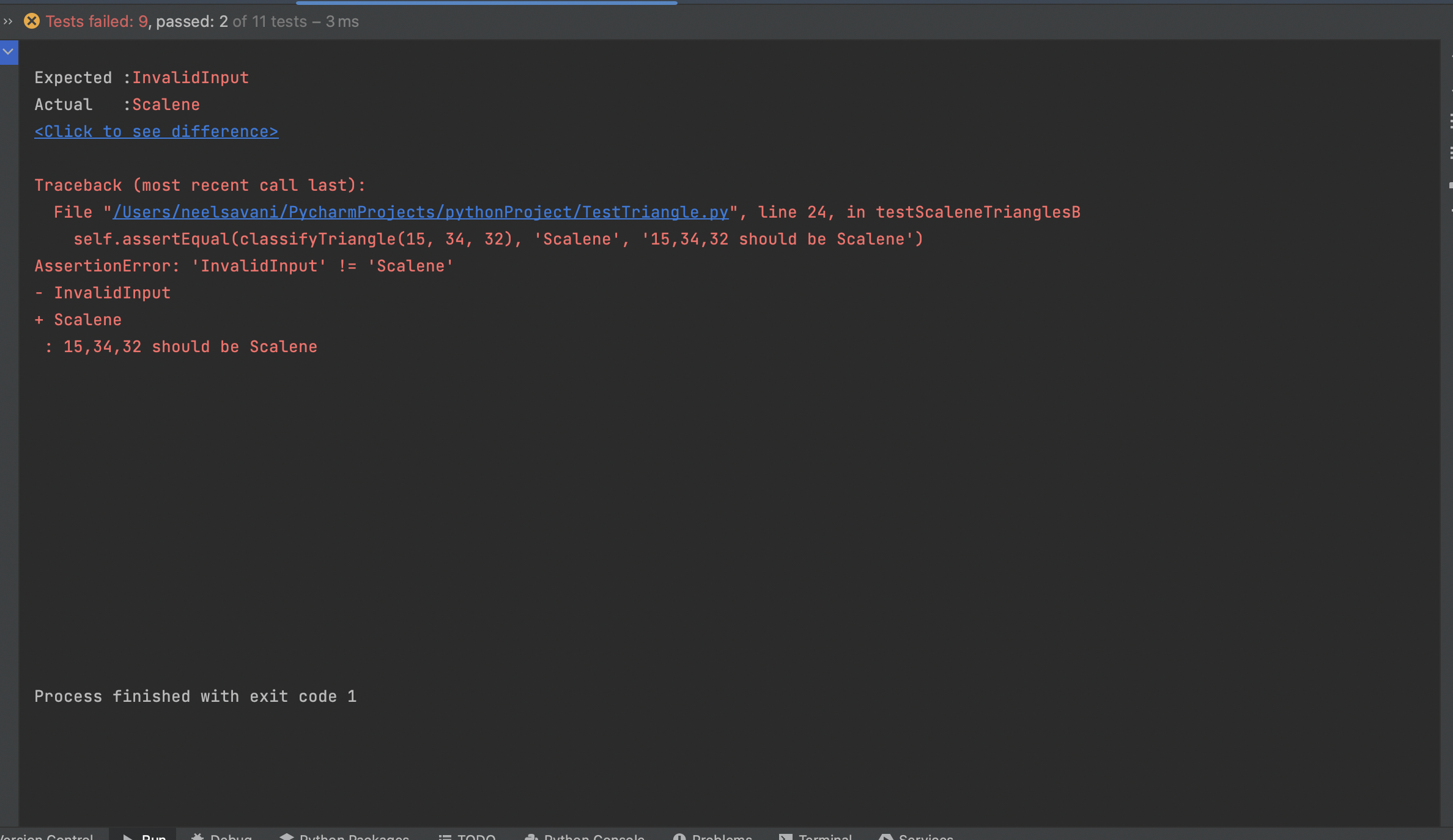












5. Include a written test report in your assignment summary with the results of running your test set against the improved implementation of classifyTriangle using  the following format:

Summary :- This is test cases report of fixed Triangle.py. While doing this assignment I have suffered with scalene triangle condition as I am giving wrong input in that so it gives NotAtriangle rather than giving me a scalene. But finally I got that I am passing wrong argument in that.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test ID** | **Input** | **Actual** **Result** | **Expected Result** | **Pass** **or** **Fail** |
| testRightTriangleA | (3,4,5) | Right | Right | Fail |
| testRightTriangleB | (5,3,4) | Right | Right | Fail |
| testEquilateralTrianglesA | (1,1,1) | Equilateral | Equilateral | Fail |
| testEquilateralTrianglesB | (5,5,5) | Equilateral | Equilateral | Fail |
| testScaleneTrianglesA | (8,11,13) | Scalene | Scalene | Fail |
| testScaleneTrianglesB | (15,34,32) | Scalene | Scalene | Fail |
| testIsosceleneTriangles | (9,9,3) | Isoscelene | Isoscelene | Fail |
| testIsNotTriangleA | (1,2,10) | NotATriangle | NotATriangle | Fail |
| testIsNotTriangleB | (5,6,20) | NotATriangle | NotATriangle | Fail |
| testInvalidInputTriangleA | (-5,-6,20) | InvalidInput | InvalidInput | Pass |
| testInvalidInputTriangleB | (5,6,201) | InvalidInput | InvalidInput | Pass |

Output:-

/Users/neelsavani/PycharmProjects/pythonProject/venv/bin/python /Applications/PyCharm CE.app/Contents/plugins/python-ce/helpers/pycharm/\_jb\_unittest\_runner.py --target TestTriangle.TestTriangles

Testing started at 4:41 am ...

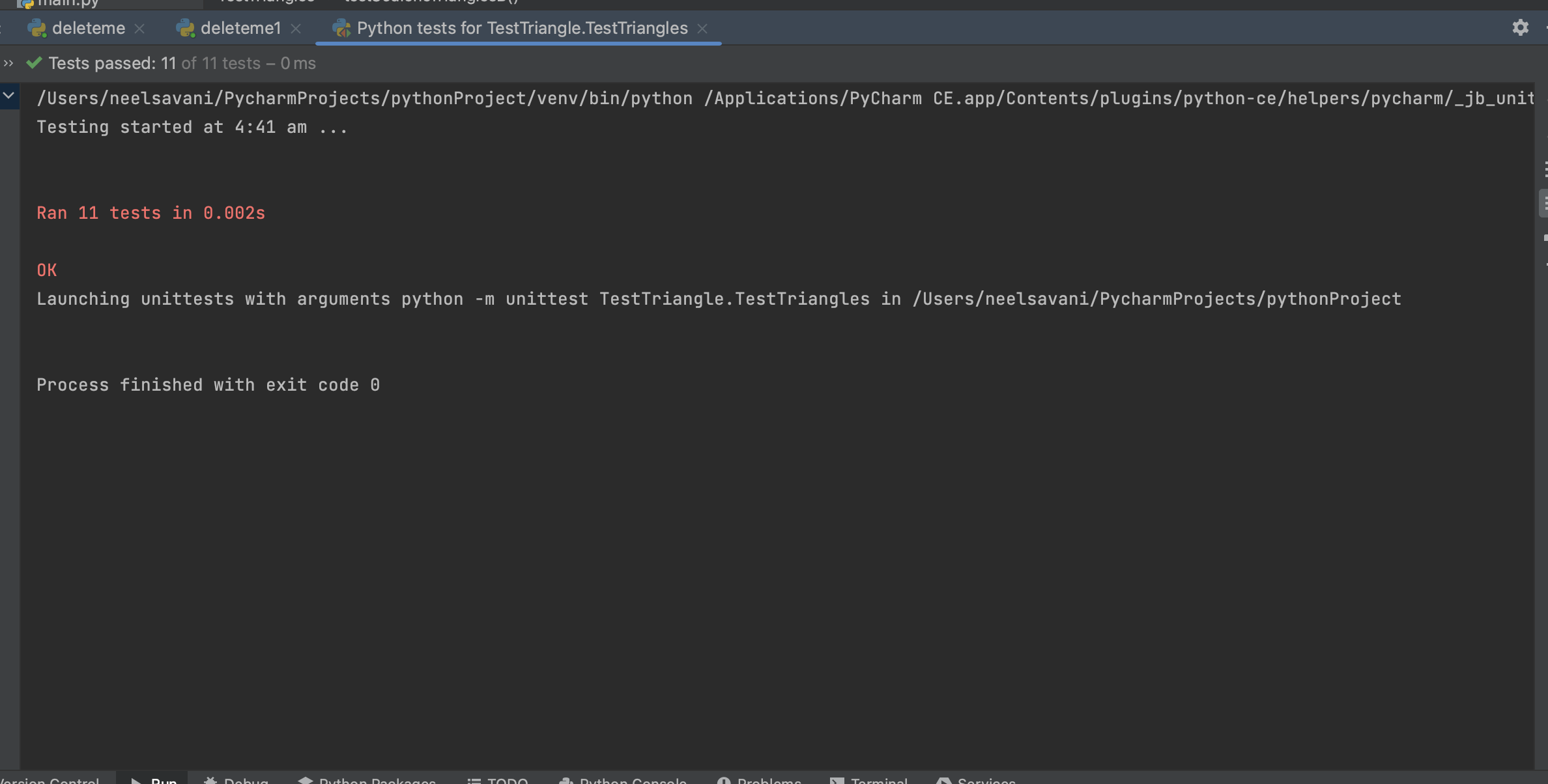
Ran 11 tests in 0.002s

OK

Launching unittests with arguments python -m unittest TestTriangle.TestTriangles in /Users/neelsavani/PycharmProjects/pythonProject

Process finished with exit code 0

These are Screenshots of Output after fix Triangle.py



6.  Your assignment summary should include a matrix as shown below in the summary results along with a description of the strategy you used to decide when you had a sufficient number of test cases.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Test Run 1 | Test Run2 | Test Run 3 |
| Tests Planned | 11 planned  -(3,4,5)-right  -(5,3,4)-right  -(1,1,1)-equilateral  -(5,5,5)-equilateral  -(8,11,13)-scalene  -(15,34,32)-scalene  -(9,9,3)-isoscelene  -(1,2,10)-notatriangle  -(5,6,20)-notatriangle  -(-5,-6,20)-invalidinput  -(5,6,201)-invalidinput | 11 planned  -(3,4,5)-right  -(5,3,4)-right  -(1,1,1)-equilateral  -(5,5,5)-equilateral  -(8,11,13)-scalene  -(15,34,32)-scalene  -(9,9,3)-isoscelene  -(1,2,10)-notatriangle  -(5,6,20)-notatriangle  -(-5,-6,20)-invalidinput  -(5,6,201)-invalidinput | 11 planned  -(3,4,5)-right  -(5,3,4)-right  -(1,1,1)-equilateral  -(5,5,5)-equilateral  -(8,11,13)-scalene  -(15,34,32)-scalene  -(9,9,3)-isoscelene  -(1,2,10)-notatriangle  -(5,6,20)-notatriangle  -(-5,-6,20)-invalidinput  -(5,6,201)-invalidinput |
| Tests Executed | All tests executed | All tests executed | All test executed |
| Tests Passed | 2 pass  -(-5,-6,20)-invalidinput  -(5,6,201)-invalidinput | 9 pass  -(3,4,5)-right  -(5,3,4)-right  -(1,1,1)-equilateral  -(5,5,5)-equilateral  -(9,9,3)-isoscelene  -(1,2,10)-notatriangle  -(5,6,20)-notatriangle  -(-5,-6,20)-invalidinput  -(5,6,201)-invalidinput | 11 pass  -(3,4,5)-right  -(5,3,4)-right  -(1,1,1)-equilateral  -(5,5,5)-equilateral  -(8,11,13)-scalene  -(15,34,32)-scalene  -(9,9,3)-isoscelene  -(1,2,10)-notatriangle  -(5,6,20)-notatriangle  -(-5,-6,20)-invalidinput  -(5,6,201)-invalidinput |
| Defects Found | 9 defects found  -(3,4,5)-right  -(5,3,4)-right  -(1,1,1)-equilateral  -(5,5,5)-equilateral  -(8,11,13)-scalene  -(15,34,32)-scalene  -(9,9,3)-isoscelene  -(1,2,10)-notatriangle  -(5,6,20)-notatriangle | 2 defects found  -(8,11,13)-scalene  -(15,34,32)-scalene | No defect found |
| Defects Fixed | 9 defects fixed  -(3,4,5)-right  -(5,3,4)-right  -(1,1,1)-equilateral  -(5,5,5)-equilateral  -(8,11,13)-scalene  -(15,34,32)-scalene  -(9,9,3)-isoscelene  -(1,2,10)-notatriangle  -(5,6,20)-notatriangle | 2 defects fixed  -(8,11,13)-scalene  -(15,34,32)-scalene | Nothing to fix |

7. Submit the name of the repo containing all of the code for this assignment

Github link:-<https://github.com/nsavani13/SSW567-ws>

Honor pledge:- “I pledge on my honor that I have not given or received any unauthorized assistance on this assignment/examination. I further pledge that I have not copied any material from a book, article, the Internet or any other source except where I have expressly cited the source.”

Author name:- Neel Hiteshkumar Savani