

Triangle with side_a=1, side_b=2, and side_c=3 make an InvalidInput
Triangle with side_a=1, side_b=1, and side_c=1 make an InvalidInput
Triangle with side_a=3, side_b=4, and side_c=5 make an InvalidInput
Triangle with side_a=8, side_b=6, and side_c=7 make an InvalidInput
Triangle with side_a=8, side_b=8, and side_c=10 make an InvalidInput

test_case_00 (test_cases.TestTrianglesBuggy)

Validate Right Triangle -PW ... FAIL

test_case_01 (test_cases.TestTrianglesBuggy)

Validate Equilateral Triangle -PW ... FAIL

test_case_02 (test_cases.TestTrianglesBuggy)

Validate Isosceles Triangle -PW ... FAIL

test_case_03 (test_cases.TestTrianglesBuggy)

Validate Scalene Triangle -CD ... FAIL

test_case_04 (test_cases.TestTrianglesBuggy)

Check Values, if a or b or c > 200 should = 'InvalidInput' -CD ... ok

test_case_05 (test_cases.TestTrianglesBuggy)

Check Values, if a or b or c <= 0 should = 'InvalidInput' -CD ... ok

test_case_06 (test_cases.TestTrianglesBuggy)

Check input value is integer else = 'InvalidInput' -CD (str) and -JT (float) ... ok

test_case_07 (test_cases.TestTrianglesBuggy)

Check that the sum of any 2 sides is greater than the 3rd side, if it fails = 'NotATriangle' -JT ...
FAIL

test_case_08 (test_cases.TestTrianglesBuggy)

Check parameter order - PW ... FAIL

=====
=

FAIL: test_case_00 (test_cases.TestTrianglesBuggy)

Validate Right Triangle -PW

Traceback (most recent call last):

File "C:\Users\Phil\Documents\GitHub\SSW567\assignment_2\test_cases.py", line 12, in
test_case_00

self.assertEqual(self.func(3, 4, 5), 'Right', '3, 4, 5 Should Be Right')

AssertionError: 3, 4, 5 Should Be Right

=====
=

FAIL: test_case_01 (test_cases.TestTrianglesBuggy)

Validate Equilateral Triangle -PW

Traceback (most recent call last):

```
File "C:\Users\Phil\Documents\GitHub\SSW567\assignment_2\test_cases.py", line 16, in
test_case_01
    self.assertEqual(self.func(1, 1, 1), 'Equilateral', '1, 1, 1 Should be Equilateral')
AssertionError: 1, 1, 1 Should be Equilateral
```

```
=====
=
FAIL: test_case_02 (test_cases.TestTrianglesBuggy)
Validate Isosceles Triangle -PW
```

```
-----
Traceback (most recent call last):
  File "C:\Users\Phil\Documents\GitHub\SSW567\assignment_2\test_cases.py", line 21, in
test_case_02
    self.assertEqual(self.func(10, 10, 8), 'Isosceles', '10, 10, 8 Should be Isosceles')
AssertionError: 10, 10, 8 Should be Isosceles
```

```
=====
=
FAIL: test_case_03 (test_cases.TestTrianglesBuggy)
Validate Scalene Triangle -CD
```

```
-----
Traceback (most recent call last):
  File "C:\Users\Phil\Documents\GitHub\SSW567\assignment_2\test_cases.py", line 25, in
test_case_03
    self.assertEqual(self.func(8, 6, 7), 'Scalene', '8, 6, 7 Should be Scalene')
AssertionError: 8, 6, 7 Should be Scalene
```

```
=====
=
FAIL: test_case_07 (test_cases.TestTrianglesBuggy)
Check that the sum of any 2 sides is greater than the 3rd side, if it fails = 'NotATriangle' -JT
```

```
-----
Traceback (most recent call last):
  File "C:\Users\Phil\Documents\GitHub\SSW567\assignment_2\test_cases.py", line 55, in
test_case_07
    self.assertEqual(self.func(11, 6, 4), 'NotATriangle', '11, 6, 4 Should be NotATriangle')
AssertionError: 11, 6, 4 Should be NotATriangle
```

```
=====
=
FAIL: test_case_08 (test_cases.TestTrianglesBuggy)
Check parameter order - PW
```

Traceback (most recent call last):

File "C:\Users\Phil\Documents\GitHub\SSW567\assignment_2\test_cases.py", line 61, in
test_case_08

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

AssertionError: 3,4,5 is a Right triangle

Ran 9 tests in 0.000s

FAILED (failures=6)

Process finished with exit code 0