

```
triangle_checker.py x
27 def test_equilateral_triangle(self):
28     self.assertEqual(classify_triangle(a: 3, b: 3, c: 3), second: "Equilateral")
29     self.assertEqual(classify_triangle(a: 3, b: 3, c: 3), second: "Equilateral")
30     self.assertEqual(classify_triangle(a: 3, b: 3, c: 3), second: "Equilateral")
31
32 def test_isosceles_triangle(self):
33     self.assertEqual(classify_triangle(a: 3, b: 3, c: 4), second: "Isosceles")
34     self.assertEqual(classify_triangle(a: 4, b: 3, c: 3), second: "Isosceles")
35     self.assertEqual(classify_triangle(a: 3, b: 4, c: 3), second: "Isosceles")
36
37 def test_scalene_triangle(self):
38     self.assertEqual(classify_triangle(a: 5, b: 4, c: 8), second: "Scalene")
39     self.assertEqual(classify_triangle(a: 10, b: 5, c: 8), second: "Scalene")
40     self.assertEqual(classify_triangle(a: 6, b: 9, c: 8), second: "Scalene")
41
42 def test_right_triangle(self):
43     self.assertEqual(classify_triangle(a: 16, b: 63, c: 65), second: "Right")
44     self.assertEqual(classify_triangle(a: 5, b: 12, c: 13), second: "Right")
45     self.assertEqual(classify_triangle(a: 8, b: 15, c: 17), second: "Right")
46
47 def test_invalid_input(self):
48     self.assertEqual(classify_triangle(a: 0, b: 4, c: 5), second: "Invalid input")
49     self.assertEqual(classify_triangle(a: 3, b: "a", c: 5), second: "Invalid input")
50     self.assertEqual(classify_triangle(a: -3, b: 4, c: 5), second: "Invalid input")
```

Run Python tests in triangle_checker.py x

Test Results 0 ms

Tests passed: 5 of 5 tests - 0 ms

Ran 5 tests in 0.002s

OK

Process finished with exit code 0