Coverage for **Triangle.py** : 76%

⌨

17 statements | 13 run | 4 missing | 0 excluded

```
1   # -*- coding: utf-8 -*-
2   """
3   Created on Thu Jan 14 13:44:00 2016
4   Updated Sep 20, 2020
5
6   The primary goal of this file is to demonstrate a simple python program to classify triangle
7
8   @author: jrr
9   @author: rk
10  @author: Srikanth Uppada
11  """
12
13  def classifyTriangle(a,b,c):
14      """
15      Your correct code goes here...  Fix the faulty logic below until the code passes all of
16      you test cases.
17
18      This function returns a string with the type of triangle from three integer values
19      corresponding to the lengths of the three sides of the Triangle.
20
21      return:
22          If all three sides are equal, return 'Equilateral'
23          If exactly one pair of sides are equal, return 'Isoceles'
24          If no pair of  sides are equal, return 'Scalene'
25          If not a valid triangle, then return 'NotATriangle'
26          If the sum of any two sides equals the squate of the third side, then return 'Right'
27
28        BEWARE: there may be a bug or two in this code
29      """
30
31      # verify that all 3 inputs are integers
32      # Python's "isinstance(object,type) returns True if the object is of the specified type
33      if not(isinstance(a,int) and isinstance(b,int) and isinstance(c,int)):
34          return 'InvalidInput';
35
36      # require that the input values be >= 0 and <= 200
37      if a > 200 or b > 200 or c > 200:
38          return 'InvalidInput'
39
40      if a <= 0 or b <= 0 or c <= 0:
41          return 'InvalidInput'
42
43      # This information was not in the requirements spec but
44      # is important for correctness
45      # the sum of any two sides must be strictly less than the third side
46      # of the specified shape is not a triangle
47      if ((a + b) <= c) or ((b + c) <= a) or ((a + c) <= b):
```

```python
48          return 'NotATriangle'
49
50      # now we know that we have a valid triangle
51      if a == b and a == c:
52          return 'Equilateral'
53      elif ((a**2 + b**2) == c**2) or ((a**2 + c**2) == b**2) or ((b**2 + c**2) == a**2):
54          return 'Right'
55      elif (a != b) and  (b != c) and (c != a):
56          return 'Scalene'
57      else:
58          return 'Isoceles'
```

« index   *coverage.py v5.3, created at 2020-10-06 00:40 -0400*