```
Triangle with side_a=1, side_b=2, and side_c=3 make an NotATriangle
Triangle with side_a=1, side_b=1, and side_c=1 make an Equilateral
Triangle with side_a=3, side_b=4, and side_c=5 make an Right
Triangle with side_a=8, side_b=6, and side_c=7 make an Scalene
Triangle with side_a=8, side_b=8, and side_c=10 make an Isosceles
test_case_00 (test_cases.TestTrianglesFixed)
Validate Right Triangle -PW ... ok
test_case_01 (test_cases.TestTrianglesFixed)
Validate Equilateral Triangle -PW ... ok
test_case_02 (test_cases.TestTrianglesFixed)
Validate Isosceles Triangle -PW ... ok
test_case_03 (test_cases.TestTrianglesFixed)
Validate Scalene Triangle -CD ... ok
test_case_04 (test_cases.TestTrianglesFixed)
Check Values, if a or b or c > 200 should = 'InvalidInput' -CD ... ok
test_case_05 (test_cases.TestTrianglesFixed)
Check Values, if a or b or c <= 0 should = 'InvalidInput' -CD ... ok
test_case_06 (test_cases.TestTrianglesFixed)
Check input value is integer else = 'InvalidInput' -CD (str) and -JT (float) ... ok
test_case_07 (test_cases.TestTrianglesFixed)
Check that the sum of any 2 sides is greater than the 3rd side, if it fails = 'NotATriangle' -JT ... ok
test_case_08 (test_cases.TestTrianglesFixed)
Check parameter order - PW ... ok

----------------------------------------------------------------------
Ran 9 tests in 0.014s

OK

Process finished with exit code 0
```