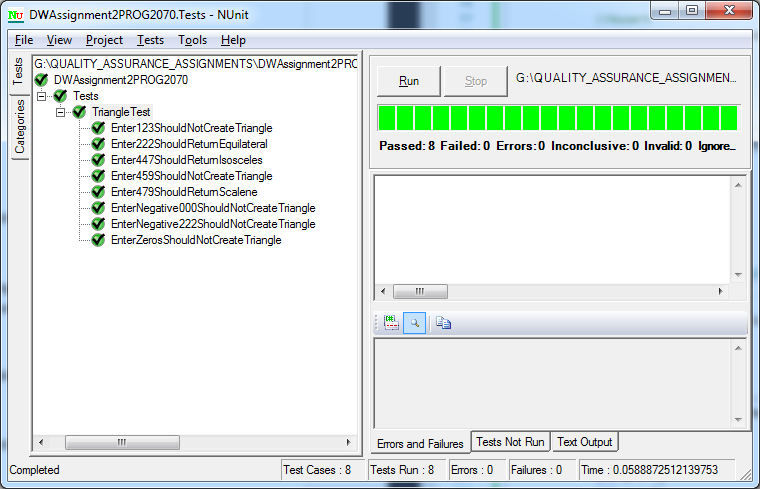
NUnit Successful Tests:



Test 1: Enter Zeros Should Not Create Triangle  
I selected zeros for the input because a triangle cannot be created with no side lengths.

Test 2: Enter 1,2,3 Should Not Create TriangleI selected 1, 2 and 3 for this test because a person uneducated in triangle geometry would probably believe this is a valid input. It is not valid because of the triangle inequality theorem which states: the sum of two sides of the triangle must be greater than the other side.

Test 3: Enter 2,2,2 Should Return Equilateral  
I selected 2, 2 and 2 for this test because when the three sides of a valid triangle are equal it is an equilateral triangle.

Test 4: Enter 4,4,7 Should Return Isosceles  
I selected 4,4 and 7 for this test because when only two sides of a valid triangle are equal it is an isosceles triangle.

Test 5: Enter 4,7,9 Should Return Scalene  
I selected 4,7 and 9 for this test because when a triangle is valid and each side is a different size, it is a scalene triangle.

Test 6: Enter 4,5,9 Should Not Create Triangle  
I selected 4,5 and 9 for this test because once again, if someone is uneducated in triangle geometry they might believe this is a valid triangle. However, 4 + 5 is not greater than 9 and therefore it fails the triangle inequality theorem.

Test 7: Enter negative 2, 2, 2 Should Not Create Triangle  
I selected -2, -2 and -2 for this test because negative values cannot create a triangle.

Test 8: Enter negative 0,0,0 Should Not Create Triangle  
I selected -0,-0 and -0 for this test because it is an odd input considering zero cannot be negative. This test makes sure the program can handle this odd input.

Control Flow Diagram for the Analyze Method:



Cyclomatic Complexity:   
*M* = *E* – ­*N* + *2P  
M* = 14 – 12 + 2  
*M* = 4

Therefore the cyclomatic complexity

Version Control Log (Git Bash):  
  
