Triangle Classification Example

Reference: Glenford Myers, *The Art of Software Testing*.

A program accepts as input three integers which it interprets as the lengths of sides of a triangle. It reports whether the triangle is equilateral, isosceles, or scalene (neither equilateral nor isosceles). Write a set of test data to test this program.

<u>**Equilateral:**</u> All three sides are the same length
<u>**Isosceles:**</u> Two sides are the same length
<u>**Scalene:**</u> Not equilateral or scalene (no sides are the same length)

Equivalence Classes
1) Valid
    a. Equilateral
    b. Isosceles
        i. All three permutations
    c. Scalene
    d. Very small triangles
    e. Very large triangles
    f. Combinations of very long and very short sides in all possible orders
2) Invalid
    a. Impossible
        i. Side lengths that are not a valid triangle
        ii. The side lengths of a valid triangle must satisfy the following: $a < b + c$, $b < a + c$, $c < a + b$
    b. One or more sides of length 0
    c. Negative side lengths
    d. Floating point side lengths
    e. Wrong number of parameters
    f. Side lengths that are out of range (i.e., bigger than max allowed value)

Boundary Conditions
1) Isosceles triangles that are almost Equilateral
2) Scalene triangles that are almost Isosceles or Equilateral
3) Triangles where the length of one side is equal to the sum of the other two (all three permutations)
4) Triangles where the length of one side is barely smaller than the sum of the other two (all three permutations)
5) Triangles where the length of one side is barely greater than the sum of the other two (all three permutations)