Suggested solution for assignment 14

The assignment:

Use the method consisting of the phases analysis, design, implementation, and testing (described in section 3.9 and the overheads) to develop a program that reads the lengths of the sides in a triangle and prints which kind of triangle it is.

Analysis:

- Should the program interact with the user?
- What is ment by: "Which kind of triangle?"
- What should the program do if the input does not represent a triangle? Should it STOP or keep promting for new input?

After a discussion with the customer, the following was agreed on:

- Yes
- "Which kind of triangle?" means: equilateral, isosceles or scalene triangle.
- It should stop with an error message.

Design 1:

Input the length of the sides in a triangle;
If the input does not describe a triangle, then print an error message,
otherwise determine and print the type of the triangle.

More detailed design:

Input the length of the sides (a,b,c) in a triangle;

Sort a, b and c so that a <= b <= c;

If a <= 0 or a + b <= c, then print "This is not a triangle",
otherwise, if a=c, then print "This triangle is equilateral",
otherwise, if a=b or b=c , then print "This triangle is isosceles",
otherwise, print "This triangle is scalene".

Implementation:

```
/*
  5.Week Exercise 14: Triangles
  Author: Lasse Lerkenfeld Jensen
*/

import cs1.Keyboard;

public class Triangles
{
  public static void main(String[] s)
  {
    //Input the length of the sides (a,b,c) in a triangle
    System.out.println("Input the length of the sides of a triangle: ");
    System.out.print("Length of side a:"); int a = Keyboard.readInt();
    System.out.print("Length of side b:"); int b = Keyboard.readInt();
    System.out.print("Length of side c:"); int c = Keyboard.readInt();

    //Sort a, b and c so that: a <= b <=c
    int temp;
    if (a > b)
    {
      temp = a;
      a = b;
      b = temp;
    }
    if (c < a)
    {
      temp = c;
      c = b;
      b = a;
      a = temp;
    }
    else if (c < b)
    {
      temp = c;
      c = b;
      b = temp;
    }

    //Determine and print the type of the triangle
    if (a <= 0 || a+b <= c)
      System.out.println("This is not a triangle.");
    else if (a == c)
      System.out.println("This triangle is equilateral.");
    else if (a == b || b == c)
      System.out.println("This triangle is isosceles.");
    else
      System.out.println("This triangle is scalene.");
  }
}
```

Test:

Cases:	Input:	Output:
a>b	3,2,6	This is not a triangle.
c<a	4,4,2	This triangle is equilateral.
c<b (and c>a)	3,5,4	This triangle is scalene.
a<=0	-2,4,5	This is not a triangle.
a+b<=c	2,3,6	This is not a triangle.
a==c	2,2,2	This triangle is isosceles.
a==b	2,2,4	This is not a triangle.
b==c	2,3,3	This triangle is equilateral.
None of those above	3,4,5	This triangle is scalene.

We can now conclude, that the program is working correctly, since the output is as expected.

Alternative solution:

Please notice, the solution above is merely a suggestion. There are other ways to solve the problem, with different approaches to the analysis and design. For example one could assume(and check) that the user input is already sorted, or have larger conditions in the if-statements.