Writing Test Cases I

February 11, 2018

Professor Badari

CMPE 187

Team 16

Raghav Gupta

Chelsea Jaculina

Yihua Li

Michael Lee

Table of Contents

1. Overview

1.1. Introduction

This is a program that asks the user to input three integer values for the side lengths of a triangle. The program will then determine if the input values are valid and whether or not the three sides form a triangle. If the input of the three sides are valid, then it is a triangle and the program will determine what kind of type of triangle (Equilateral, Isosceles, Scalene) it is. If the input for the side lengths is invalid, it is therefore not a triangle and will provide feedback to the user stating it is not.

The program continuously asks the user if they want to continue running the program. The user should then reply with a ‘y’ for yes or a ‘no’ for no.

1.2. Purpose

The purpose of test cases is to make sure the system is satisfies all requirements for the design of the program, check for any bugs/errors, or for any possible corner cases before the program is released. Although there is no bug-free application, creating a multiple number of test cases helps increase code coverage of the program, which lowers the chance to detect bugs/errors.

2. Oracle

2.1. Environment

The hardware that we used for programming and testing is on a MacBook Pro.

The software that we used for programming and testing is the Eclipse Java EE IDE for Web Developers, Version Oxygen. We also implemented certain testing methods during our process of generating the test cases. We incorporated unit testing when we first wrote the basic test cases to check whether the triangle was an equilateral, isosceles, or scalene. After checking that unit, we moved on to generating test cases for negative numbers and then for invalid inputs made by the user.

2.2. Scope

This program will take in three integer values, representing three sides of the triangle. If those three sides form a valid triangle, then the program will determine what type of triangle it is. The definition of a valid triangle is that it has three sides, the length of the sides are non-negative, and that the sum of two sides are greater than the third side. An equilateral triangle has all of their lengths equal to one another. An isosceles triangle has at least two sides that are equal length and a scalene triangle will have three different side lengths.

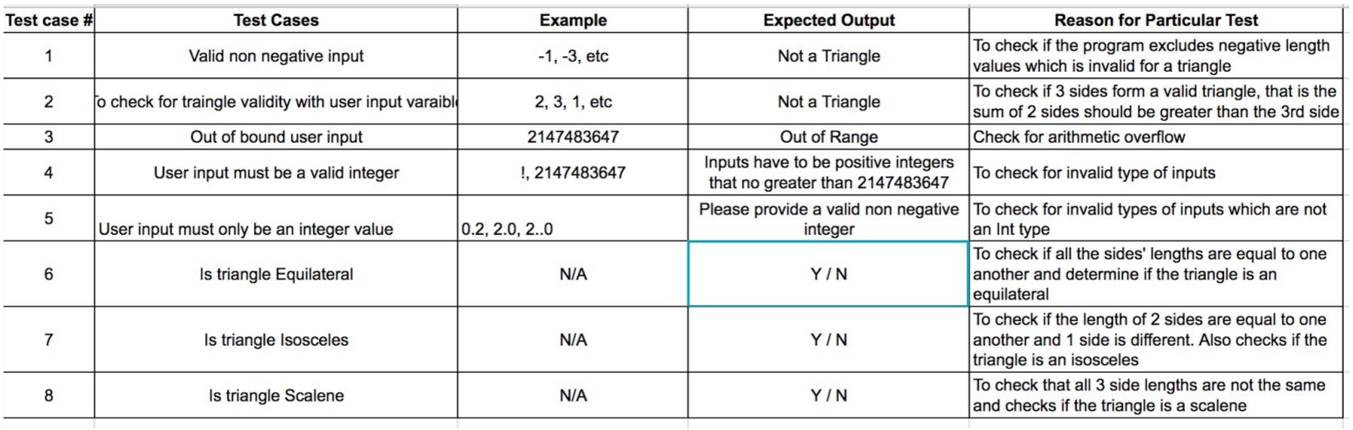
Since the length of sides of triangles has to be larger than 0, this program will only accept integers that are greater than 0. In addition, since the program we developed is implemented in the Java programming language, we need to check if the integer that the user inputs is valid. In the Java programming language an integer has a max value of 2^32 (2,147,483,647). Therefore, the max value of the sum of any two sides is 2,147,483,647.

If the user decides to input another data type such as a String, char, double, the program will terminate as it will not be acceptable.

3. Test cases

As mentioned beforehand, test cases help discover and point out the bugs or errors in the program. Every test case needs to have some sort of input and expected outputs. In our program our input values are integers and are expected outputs are of String. This String output determines if the values produce a valid triangle and the type of triangle.

In Figure 1, we can see a table consisting of 8 test cases that our team has written. Our team then developed a simple Java program keeping these test cases in mind.



Test case #	Test Cases	Example	Expected Output	Reason for Particular Test
1	Valid non negative input	-1, -3, etc	Not a Triangle	To check if the program excludes negative length values which is invalid for a triangle
2	o check for traingle validity with user input varaibl	2, 3, 1, etc	Not a Triangle	To check if 3 sides form a valid triangle, that is the sum of 2 sides should be greater than the 3rd side
3	Out of bound user input	2147483647	Out of Range	Check for arithmetic overflow
4	User input must be a valid integer	!, 2147483647	Inputs have to be positive integers that no greater than 2147483647	To check for invalid type of inputs
5	User input must only be an integer value	0.2, 2.0, 2..0	Please provide a valid non negative integer	To check for invalid types of inputs which are not an Int type
6	Is triangle Equilateral	N/A	Y / N	To check if all the sides' lengths are equal to one another and determine if the triangle is an equilateral
7	Is triangle Isosceles	N/A	Y / N	To check if the length of 2 sides are equal to one another and 1 side is different. Also checks if the triangle is an isosceles
8	Is triangle Scalene	N/A	Y / N	To check that all 3 side lengths are not the same and checks if the triangle is a scalene

Figure 1. Test Cases for Triangle Java Program

4. Results

Our team was able to implement every one of the 8 test cases that we wrote into source code. The user was able to test as many triangles they wanted and receive feedback on whether or not the it was a triangle and the type of triangle.

5. Lessons Learned

Overall as a team, we learned how to write test cases for a software program, discovered what’s the importance of test cases, and developed code based off the test cases that we generated.