James Valles
Part II of Assignment #1
SE 433


a. **Name of my Partner:** David Bergunder

b. For each test case, indicate the actual output of the program and whether the test case pass or fails. << Please refer to excel spreadsheet.

c. The only weakness I discovered in my partner's program is when I entered no input and pressed return, the program didn't throw an error. This could be correct, or a test case failure based on what is assumed to be correct.

I assumed and my program implements the following when addressing this test case. When you press enter, whatever is inputted will be evaluated. So, if you enter nothing and press return, entering nothing will result in a NumberFormatException. The homework requirement didn't specifically address this. Although, it does say "a. The user enters 3 values at a prompt and presses return." So based on interpretation, one could assume that the program will only accept 3 values at the prompt. And when you press return whatever is entered, including nothing, will be evaluated.

Other than that, I ran a range of test cases, and my partner's program passed them all.

d. Developing my test suite was a bit more involved than I originally thought. At first, I forgot to test the input string 'exit' and ensure that the program terminates. Furthermore, I didn't test for MAX integer + 1 input, ie: 202020202002, which results in a NumberFormatException even though at first glance it like an integer. But, should be tested. I am glad I included those tests in my test suite. I also found it important to run the same type of tests with different values. For example, checking for 0 lengths, during development when I entered 0 0 0, I got equilateral, which is clearly incorrect. Because I was developing my tests while writing the program. Performing these test allowed me to identify any early bugs during development which lead to improved and correct code.

e. I found this assignment to be of tremendous value. First, it allowed me to really think about all of the test cases that could possibly create an error. I ended up with 29 test cases, although, I believe this number could be reduce. But, I felt it was better to over test since I had the time and am learning. Writing down the test case using a spreadsheet kept my thoughts organized and I believe the process would help write automated tests. So, I really appreciate the experience. It thought me how to keep my thoughts organized, allowed me to brainstorm all possible test cases, helped me discover bugs in my code during development. It also emphasized the importance of requirements (the checklist). The more ambitious the requirements, the harder it is to identify what is correct or incorrect when testing.