Kyle De Laurell

Understanding:
I learned that it is not just understanding the language or code C++. You have to truly understand the problem and what your end goal is. I learned this in the group portion because we were brainstorming how to solve the problem we knew all the different loops and how to write them, but our biggest struggle was that we didn't understand which one to use and how to use it to accomplish our end goal. We finally figured it out, but only after a deep discussion of what we were actually trying to accomplish.

Testing Plan:
My only tests that did not work out as planned was on problem 3b. I outputted to the display every number that was being read in and the sum after that number was added in. It kept skipping every other number that it inputted. Based on this, after some deep reading and looking at the examples, I realized that I was reading two lines from the text file every iteration in my loop so that is why it skipped every other number. I did not need to read in a line from the text file because it was already being read in in my conditional statement for my while loop. I feel very comfortable that my planned tests were complete. I tried to think of every conceivable way to break my program within the constraints of the assignment.

Design:
Aside from my initial design from above where I inputted two lines from the text file in 3b. The only other missing piece that came up was on 3c when I was writing the last statement where I state how many guesses it took to guess the correct number. I realized that I forgot to create a counter variable to count the number of guesses it took to guess the correct number so I had to add the variable and insert it into my program.

Implementation:
My implementation problem was once again in 3b when I realized that the sum that was outputted to sum.txt was not correct because I was reading two lines every iteration of my loop. I was able to solve the problem by re-reading the section on files for data storage and it lead me to my answer.

Improvement:
I think that I could do a better job of laying out the groundwork or pseudocode for my program before diving right into the coding aspect of the assignment. I feel that by doing this it will help me to better visualize any gaps that I have in my logic and identify areas where I could possibly improve my code before I actually start coding.