Marcus Plumley

CS133c

10-18-19

Assignment – 2 README

**Design:** The design of this project was fairly simple since we are not dealing with a lot of data, but the loops needed to be structured well since we want to keep the option of playing the game open. I used two while loops for the game function because one while loop kept the guessing loop going and the other while loop kept the game loop going.

The guessing loop needed to have three different if statements for the three possibilities: guessed too low, guessed too high, or exact. The lower and higher guesses adjust the bounds given to the user and adjust the guess limit. The correct guess conditional notifies the user and breaks the guess loop.

Once the guess loop is broken, the game loop prompts the user on whether they wish to continue the game loop or not. After that, the main function is very basic and just calls the game function with some preset values for bounds and guess limit.

**Testing:** The testing for this program must check for all three guess outcomes for the expected result. All three guess situations worked with the program if the input was proper. I did not have input validation for type or magnitude, so if letters are entered, then the program will fail. In addition, the bounds will fail if the guessed value is either higher or lower than the bounds calculated.

**Reflection:** The lessons that we worked on in week 2 were all about the basics and input/output. The basics that we focused on were loops and conditionals and using that foundation and what we learned with input/output, we are able to begin the guessing program. I learned during this problem that organization of loops and functions is very important and can really save you some time if you design it beforehand. I had redesign because during implication of my first design, I was unable to foresee a problem with how I organized the loops. Finally, this project seems like a prerequisite to the next which is guess a word rather than number.