**CS 31 Project 5 Report**

**Notable obstacles I faced:**

1. It was hard to identify how many planets and stars were in the prompt entered by the user when characters that were planets and stars were repeated.
2. Debugging the code in order to find minor bugs was incredibly hard as they were not easily visible.

**Pseudocode:**

Function runOneRound:

Check if number of words are less than or equal to 0 and if the position of the secret word is outside of the interesting elements in the array.

Repeat following steps,

Accept probe word as input from the user and store it as a C String.

Check if the probe word only contained 4 to 6 lower case alphabets and behave accordingly. Display an error message if it doesn’t meet the conditions and repeat previous step.

Repeatedly parse through the words in the array of C Strings and check if the prompt is a valid word.

Repeatedly, parse through the characters in prompt.

Repeatedly, in a nested manner, parse through characters in secret word.

Repeatedly, in a nested manner, parse through elements of the array, index, to see if the current character in the secret word being parsed has already been identified to be a planet or a star.

If the character has not previously been discovered to be a planet or a star, and the character in the prompt is the same as the character in the secret word and is in the same position, it is a star.

The position of the character is stored in the index array if it is a star

If the character has not previously been discovered to be a planet or a star, and the character in the prompt is the same as the character in the secret word but is not in the same position, it is a planet.

Planets can be stars if the same character is repeated in another position. Thus, parse through the remaining characters in the prompt and secret word to see if the planet is a star in another position.

The position of the character is stored in the index array if it is a planet or a star

After the prompt has been parsed completely,

Display the number of stars and planets in the prompt if the prompt is not the same as the secret word

If the prompt is the secret word, return the number of tries the user took to figure out what the secret word was. The function is thus terminated.

Function main:

Create an array with the capacity to hold 9000 C strings with a maximum length of 7 characters.

Call the function, getWords in utilities.cpp to store the number of words in a text file in the array of C strings. Store the number of interesting words in a variable.

If the number of interesting words is less than one, terminate the game

Otherwise, do the following.

Accept the number of rounds the user wants to play.

If the number of rounds entered is less than one, display an error message and terminate the game.

Otherwise, do the following repeatedly as many times as the number of rounds specified by the user.

Generate a random integer to randomize secret word for the round.

Call the function runOneRound to run the round and store the number of tries the user takes to figure out what the secret word is.

If it is the first round, initialize variables holding the maximum and minimum number of tries to the number of tries in the first round.

Otherwise compare the number of tries in the current round to the previous maximum and minimum and change values accordingly.

Calculate the average number of tries in the game.

Display the number of tries taken in the current round.

Set the double precision to 2 decimal places and display the average, maximum and minimum number of tries so far in the game.