Project plan: Project 1.a Jared Gaskin 932724470

Testing Plan

Description of Test	Expected Test Results
Assume program has selected the word "language" as the puzzle word User inputs the following: 'a' [Enter] …… 'g' [Enter] …… 'y' [Enter] …… 'v' [Enter] …… 'l' [Enter] …… 'n' [Enter] …… 'u' [Enter] …… 'e' [Enter] …… 'n'[Enter] ……	Assume program queries for input between each line "Welcome to Hangman" - - - - - - - - Guessed letters: " " - a - - - a - - Guessed letters: "a" - a - g - a g - Guessed letters: "a g" - a - g - a g - Guessed letters: "a g y" - a - g - a g - Guessed letters: "a g y v" l a - g - a g - Guessed letters: "a g y v l" l a n g - a g - Guessed letters: "a g y v l n" l a n g u a g - Guessed letters: "a g y v l n u" l a n g u a g e Guessed letters: "a g y v l n u e" "You Win! It took you 8 guesses" "Play Again? [Y/N]"

Description of Test	Expected Test Results
Assume program has selected the word "soccer" as the puzzle word User inputs the following: 'a' [Enter] …… 's' [Enter] …… 'g' [Enter] …… 'y' [Enter] …… 'c' [Enter] …… 'o' [Enter] …… 'l' [Enter] …… 'r' [Enter] …… 'e' [Enter] …… 'n'[Enter] ……	Assume program queries for input between each line "Welcome to Hangman" - - - - - - Guessed letters: " " - - - - - - Guessed letters: "a" s - - - - - Guessed letters: "a s" s - - - - - Guessed letters: "a s g " s - - - - - Guessed letters: "a s g y" s - c c - - Guessed letters: "a s g y c" s o c c - - Guessed letters: "a s g y c o" s o c c - - Guessed letters: "a s g y c o l" s o c c - r Guessed letters: "a s g y c o l r" s o c c e r Guessed letters: "a s g y c o l r e" "You Win! It took you 9 guesses" "Play Again? [Y/N]"

Description of Test	Expected Test Results
Assume program has selected the word "soccer" as the puzzle word User inputs the following: 'a' [Enter] …… 'a' [Enter] …… 's' [Enter] …… 's' [Enter] …… 'g' [Enter] …… 'g' [Enter] …… 'c' [Enter] …… 'c' [Enter] …… 'o' [Enter] …… 'e' [Enter] …… 'r' [Enter] …… 'y' [Enter] …… Program selects a new word: "computer" 'c' [Enter] …… 'b' [Enter] …… 'o' [Enter] …… 'm' [Enter] …… 'a' [Enter] …… 'p' [Enter] …… 'u' [Enter] …… 't' [Enter] …… 'e' [Enter] …… 'r' [Enter] …… 'n' [Enter] ……	Assume program queries for input between each line "Welcome to Hangman" - - - - - - Guessed letters: " " - - - - - - Guessed letters: "a" - - - - - - Guessed letters: "a" s - - - - - Guessed letters: "a s" s - - - - - Guessed letters: "a s" s - - - - - Guessed letters: "a s g" s - - - - - Guessed letters: "a s g" s - c c - - Guessed letters: "a s g c" s - c c - - Guessed letters: "a s g c " s o c c - - Guessed letters: "a s g c o" s o c c e - Guessed letters: "a s g c o e" s o c c e r Guessed letters: "a s g c o e r" "You Win! It took you 7 guesses" "Play Again? [Y/N]" "Welcome to Hangman" - - - - - - - - Guessed letters: " " c - - - - - - - Guessed letters: "c" c - - - - - - - Guessed letters: "c b " c o - - - - - - Guessed letters: "c b o" c o m - - - - - Guessed letters: "c b o m" c o m - - - - - Guessed letters: "c b o m a" c o m p - - - - Guessed letters: "c b o m p" c o m p u - - - Guessed letters: "c b o m p u" c o m p u t - - Guessed letters: "c b o m p u t" c o m p u t e - Guessed letters: "c b o m p u t e" c o m p u t e r Guessed letters: "c b o m p u t e r" "You Win! It took you 7 guesses" "Play Again? [Y/N]"

Design Specifications (taken directly from project requirements page):

Write a program that lets the user play hangman. It should have an array of ten strings from which it randomly selects one. It should initially print a string of hyphens that is the same length as the string. The program should then prompt the user to pick a letter. If the letter matches any characters in the string, the hyphens in the matching positions should be replaced with that character. Whether or not it's in the string, the letter should be added to the list of letters chosen so far, which should also be printed out for the user. The program should continue asking for a letter and then printing the updated string and characters chosen so far until the string is complete. At that point, the program should congratulate the user and tell them how many guesses it took.

Program Design Pseudocode:

```
Begin program
loop(while program is running)

        create list of possible words
        create list of guessed letters
        create a place to store number of guesses
        puzzle word = random word from list of words
        puzzle = as many hyphens as there are letters in the puzzle word

        loop(while game has not been won)

                print out puzzle

                Ask the user to guess a letter

                If (guessed letter in puzzle word and not already guessed)

                        …Show the user the puzzle with the guessed letter now filled in
                        ….Add the guessed letter to a list of guessed letters
                        …Increase the number of guesses by 1

                If (guessed letter is not in the puzzle word and not already guessed)

                        …Add the guessed letter to a list of guessed letters
                        …Increase the number of guesses by 1

                Show the user the list of guessed letters

                If (puzzle being shown is the full puzzle word).

                        …Exit this loop by stating that the game has been won
                        …Show how many letters the user guessed
        end loop

        loop(while game has been won)

                Ask the user if they want to play the game again
                If(the user responds yes)

                        …exit this loop

                If (the user responds no)

                        …exit the program
        end loop


end loop
End Program
```