Preethika Kiruveedula

204912895

Project 5 Report

1. Some of the many obstacles I overcame are: understanding the requirements of the spec, figuring out how to reference the getWords function in terms of its parameters, the extensive use of C-strings, successfully implementing the randInt function and lastly my biggest issue was when to return the sentence “I don’t know that word.” I had a somewhat difficult time formatting my code in such a way that it addressed all the conditions of the spec without being extremely repetitive or using C-strings.

int main()

{

declares each element of the array as being able to hold a C string of length up to 6 letters

declares an integer equivalent to the return of getWords

if the return value of getWords is less than 1

terminates the game

Asks how many rounds the user wants to play

If the number of rounds is not positive

Terminates the game

Goes through round one

Declares an integer equivalent to the position of the secret word in the array

Outputs the length of the secret word

Declares an integer equivalent to the return value of runOneRound

Initializes the average as the score of the first round

Initializes the minimum as the score of the first round

Initializes the maximum as the score of the first round

Outputs the amount of tries it took to go through the round

Outputs average, minimum, and maximum

Repeatedly do the same thing above for the rest of the rounds

Updates the minimum, maximum, average, scoreOfOneRound, and the total

score of the rounds

}

Int runOneRound()

{

If the number of the words in the array is negative

Return -1

If the position of the secret word in the array is less than zero or greater than the

number of words

return -1

initialize score

infinitely

declare probe word and prompt user

if the length is less than 4 or greater than 6

output that your probe words must be 4 to 6 lower case

continue

declare a bool notLower as false to check if all characters are lower case

if a character in the probe word is not lowercase bool will be true and output

must be four to six lower case letters

declare a bool found as false

repeatedly

compare the probe word with elements of the array

if probe word is not in array bool is false

compare the probe word with elements in the array

if probe word is in the array bool is true

if found is false

output I don’t know that word

if comparison between probe word and secret word is greater than or less than

zero

declare stars and planets

increment one to score

declare an array of stars

repeatedly

if probe word and secret word have the same character in the

same position

stars++ and add the character to the array of stars

declare an array of planets

repeatedly

if bool is star or bool is planet is true

continue

repeatedly

if a character in the probe word is equal to some

character in the secret word

planets++ and add the character to array of planets

output number of stars and planets

}

bool isStar(character in probeWord, array of stars)

{

if the length of the array of stars equals 0

false

repeatedly

if a character in the array of stars is equal to a character in the probe word

true

Return false

}

Bool isPlanet(character in Probe Word, character in list of planets)

{

If the length of the array of planets equals 0

return false

Repeatedly

If a character in the array of planets is equal to a character in the probe word

true

return false

}