Maeneka Grewal -204901948

CS31

Project 5

a. The biggest obstacle in this code by far was writing the “stars and planets” function. First, the program had to find letters matching in the same places, which were ‘stars’. This was a simple double-for loop, and was quite easy. The trick was then making sure that anything that was already as ‘star’ could not be a ‘planet’. My original idea was to create a c-string of stars, and if a planet matched one of the stars, then it could not be counted as a planet. This didn’t work, because sometimes there is more than one pair of the same letter. Instead, I decided to create two new c-strings that replaced each star with a space. Then, for-loops could be used to check to see if there were planets in these new c-strings. I then had to make sure that letters being used as planets could not be used as planets again. So I had to make a new array of c-strings to do this. Pretty much, this function was a series of me writing code, testing it, finding test cases that caused the wrong output, editing the code, fixing those original errors but creating new ones, and starting over. Eventually, I got it as correct as I could make it. I think there are still a lot of errors.

b.

MAIN:

-create c-string of ‘wordlist’ with 7 char max and that can hold 9000 words

-call to getWords and set that to numWords

-if numWords is less than zero or too big for wordlist

return the error message “no words were loaded” etc.

-receive input for how many rounds should be played and store in rounds2play

if not positive, return “Must be positive” error message

-create variables for total score, max score, min score, and average score

-for loop, once for each round

print out Round number

get random number and random word, set random word to the ‘secret’ word

prints secret word length

sends everything to runOneRound and sets return value to score

if score is one

corresponding statement

else

corresponding statement

calculate total score, average, and checks to see if latest score is the new max or min

prints latest stats

CHECK LETTERS:

-for loop to loop through c string chars

if one of the chars is not a lower case letter, return false

-return true

RUN ONE ROUND:

-ask for first probe word

score is 1

while the probe word doesn’t equal the secret word

if the probe word isn’t proper length

return error message

ask for new probe word and continue to top of while

if the probe word has chars that aren’t lowercase letters

return error message

ask for new probe word and continue to top of while

if the probe word isn’t in word list

return “I don’t know that word” message

ask for new probe word and continue to top of while

if the probe word is valid

stars and planets are set to 0

send probe word, secret word, stars, and planets to starsPlanets function

print out number of stars and planets

if stars isn’t equal to length of secret word (match)

ask for new probe word

increment score

-return the score

STARS PLANETS:

-stars and planets are 0, create new char array called notStarTarget and notStarGuess and copy target and guess into them

-if target is shorter or they are equal

loop through target

if the two chars in guess and target match in the same position

stars is incremented

the chars in notStarTarget and notStarGuess are replaced with spaces

-if guess is shorter

loop through guess

if the two chars in guess and target match in the same position

stars is incremented

the chars in notStarTarget and notStarGuess are replaced with spaces

-new array called alreadyPlanet that holds planets

-if all the characters weren’t stars

loop through notStarTarget

loop through notStarGuess

if the chars at any position match and are not spaces

check to make sure the character is isn’t already a planet

if it is not already a planet, increment planets and add to alreadyPlanet array