1. There were many obstacles I overcame during this project. Before this project, I was not very familiar with c strings, but now I am more comfortable with c strings. One problem that I encountered in this project was the planets portion of the program. First, the planets would output an incorrect number because it would increment planets it there was multiple same letters in the word. To solve this problem, I used a break when two of the letters would be a planet to break out of the inner loop. I encountered another problem with the planets because the planets would also be incremented for the characters even when the character was a star. For this problem, I had to create temporary c strings for the secret and probe word and eliminate the index at which there is a star and copy this new string into the temporary c strings and use these strings for the planets code.
2. …

Eliminates the index from the given c string

Initializes a c string, arr of length 6

Repeatedly…

If the number is not the index

The character in the given c string is put in arr

J is incremented

Copies arr characters into the given c string

…

Runs one round of the game

Initializes secret and probe c string with length 7

The result is initialized to 0

Answer, notLowercase, and notInList are bool variables to test cases

If the index of the secret word is less than 0 or more than number of words

Result is -1

Else

Tries, the number of attempts is initialized to 1

The index, stars, and planets are initialized to 0

Copy the word from the list to the secret c string

Prints out the length of secret word

While the answer is not right

Initializes temporary probe and secret

Copies secret into the temporary secret word

Reinitializes planets, stars, and index to 0

Asks user for the probe word

Copies the probe to the temp probe

Repeatedly…

If the character is not lowercase

Not lowercase is true

Break out of loop

Else

Notlowercase is false

Repeatedly...

Copies a word from the list to temp c string

If the probe and temp c string are not the same

notInList is true

else

notInList is false

breaks out of the loop

if the length of probe is not 4-6 letters

outputs that the word is not 4-6 lowercase letters

else if the probe is not all lowercase

outputs that the word is not 4-6 lowercase letters

else if the probe is not in the list

outputs that i don’t know that word

else

if the secret and probe words are not the same

while the element is not the null byte

if two characters are equal at the same index

stars adds 1

eliminates index from temp probe

eliminates index from temp secret

index adds 1

repeatedly…

repeatedly…

if indexes are equals

planets adds 0

else if the characters are equal

planets adds 1

break out of the loop

tries adds 1

outputs number of stars

outputs number of planets

else

answer is false

result equals number of tries

returns the result

…

Main method

Initializes a 2-D c string for the wordlist

Initializes max words in the list

Initializes nwords by calling getwords

If nwords is less than 1 or greater than max words

Outputs that no words were loaded

Else

Outputs how many rounds do you want to play

User inputs an integer rounds

Initializes x to 1

Initializes totalTries, average, min, max to 0

While x is less than rounds

Wordnum is picked randomly from function randInt

Outputs round number

Num is initialized by calling function run one round

If num is -1

Outputs program failed

If the num is 1

Outputs you got it in 1 try

Else

Outputs how many tries the user gets

Totaltries adds the num

Average is totaltries divided by x

If x is 1

Min is num

Else if num is less than min

Min is num

Else

Min adds 0

If x is 1

Max is num

Else if num is less than max

Max is num

Else

Max adds 0

Outputs the average, minimum, and maximum

X adds 1

…