1. Obstacles
   1. I couldn't figure out how to check if it was a valid probe, so I just created a different function to check for that and broke it down into pieces, making it easier to tackle.
   2. My count for gold and silver was counting repeats and I couldn't figure out how to fix this. I went for office hours and then decided just to replace every char that was already counted as an uppercase letter so then no repeats would occur.
   3. I forgot to add cin.ignore after asking for the number of rounds so the first probe attempt would always run without actually taking input from the user. I fixed this by simply adding cin.ignore after taking in the number of rounds.
   4. When I ran the sanity checker I realized the formatting on my max and min output lines was incorrect and needed to be changed.
2. Pseudocode

**checkRightAnswer**

**if** probe length **and** right answer length is the same

repeatedly

check to see every **char** is the same in probe **and** answer

**return** **false** **if** **not** equal

**return** **true** **if** the words are the same

**validProbeWord**

**if** input is **not** between 4 **and** 6 chars

output error message

copy first 6 chars from input to probeWord

set index 6 in probeWord as '\0'

**for** every **char** in the index

**if** **not** alphabet **or** **not** lowercase

output error message

**return** **false**

**for** every word from the list of words

**if** probe word doesn't match

error message

**return** **false**

**return** **true** **if** all conditions passed

**countForGoldAndSilver**

create a copy of rightWord

**if** checkRightAnswer **false**

**for** every **char** in the probe word

**for** every **char** in the rightWord

**if** probe **char** is the same as rightWord **char**

**if** the chars have the same index

add to the gold count

set rightWordCopy at **this** index to 'Z'

set probe at **this** index to 'K'

**else**

add to the silver count

set rightWordCopy at **this** index to 'Z'

set probe at **this** index to 'K'

output number of golds **and** silvers

**playOneRound**

**if** nWords is negative **or** wordnum is negative **or** wordnum is greater than **or** equal to nWords

**return** -1

**while** checkRightAnswer is **false**

**do**

ask **for** probe

**while** (**not** a validProbeWord)

countForGoldAndSilver

add to score

**return** score

**main**

getWords to fill array listOfWords

**if** the number returned is less than 1

print error message

**return** 1

ask **for** number of rounds

**if** number of rounds is negative

print error message

**return** 1

**for** every round played

rand is a random integer generated by randInt

print the length of the array of chars at listOfWords index rand

numTries is the score from playOneRound

print numTries

add numTries to array of scores

max **and** min equal to numTries

**for** every round played yet

check **if** any score is greater than max

set max to said score

check **if** any score less than min

set min to said score

add score to total

print average score, max **and** min