- A brief description of notable obstacles you overcame.
    - I struggled calling another function in the loop function. I first thought the recursion was enough, but I needed to call printPermutations to finish the loop and provide all the permutations.
    - Additionally, I often got stack overload when I tested long words. They would take a long time to load. I tried to fix this by adding a function that read in only the words of the same length from dict to be compared and that shortened the time.
- A list of the test data that could be used to thoroughly test your program, along with the reason for each test. You do not have to include the results of the tests, but you must note which test cases your program does not handle correctly
    - vocabularyCreator on an empty file
    - vocabularyCreator on the words.txt file
    - vocabularyCreator on file size larger than 30000
    - potentialSequences
    - adding permutations of “abcd” into the dict file and checking that the results do not go past 20, even with more than 20 permutations
    - Using outcomeDisclosure to print everything