CS109 Assignment 2 - Wordle Solver

Due date and time: Thursday February 29, 11 pm

50 Points: 30 based on the performance of your program compared to other student submissions, 10 for your program writeup explaining your approach (a comment in your WordleSolver class), and 10 for program hygiene. Follow the program hygiene guidelines as explained in PEP 8. https://peps.python.org/pep-0008/

Starter File: **`wordle_solver.py`** Available on Canvas. You must use this starter file. It handles reading the files with possible words. Note, you are completing a class named WordleSolver, that attempts to solve a Wordle problem. **Note, for this assignment we are using words with length 7 instead of 5. Also, the guesser has 7 chances instead of 6. And finally the words are composed of lower- case letters.**

Dictionary File: Available on Canvas. **`all_words_7.txt`** file contains all the words that are considered valid words If you need help determining the current working directory, import the os module to your program with the **`import os`** statement and execute the following statement in your program: **`print(os.getcwd())`**

Tester File: Available on Canvas. **`wordle_test.py`**. This is the tester program we will use when comparing your program to the suggested solution and other students' solutions. Note, the random seed and secret word may be changed.

Submission: File name must be: **`wordle_solver.py`** For this assignment turn in your **`wordle_solver.py`** program to Canvas Assignment 2. We are **not** using GradeScope this time.

Your program may use features up to and including those from Python version 3.7, the version installed on the CS department public lab machines. If you use features from versions of Python beyond 3.7 your program may not work when graded.

The program must be your own work. You may not copy code from any source. Do not look up or copy any specific Wordle solving algorithms. Do not use a LLM or generative AI to create your algorithm or program.

The algorithm you create must be your own. Copying code from another source or failure to create your own algorithm is cheating and will result in an academic

dishonesty case filed with the Student Conduct and Academic Integrity. https://deanofstudents.utexas.edu/conduct/

Program: Complete the WordleSolver class. The only method the tester program shall call is the `getGuess(self, feedback)` method.

You may add whatever instance variables and methods to assist.

In this version of Wordle words are 7 letters in length (instead of the standard 5) and the player playing the game, your WordleSolver, has 7 guesses (instead of the standard 6) to solve the puzzle. **Note, unlike assignment 1, the words consist of lower-case letters, not uppercase letters. Write your solution accordingly.**

The WorldeSolver has an instance variable, `__all_words`, that stores all possible words. The words that can be chosen as the secret word is unknown other than being a subset of `__all_words`.

Your goal is to create an algorithm that tries to solve the Wordle puzzle in the fewest number of guesses. A major part of your grade for the assignment will be how your algorithm compares to those submitted by other students.

Another major part of your grade is an explanation of your approach. How was it developed and how does it work? What are the results of your own testing? This is worth 15 out of 50 points. Therefore a few scant sentences will receive a low grade. The expectation is for a well thought out, complete explanation of your approach. Think of it as a lab report.