Short Description

Word Puzzle is a game where the player will try guessing a secret word, one letter at a time. The player has a limited number of four guesses. Each time the player guesses a letter that is not part of the secret word or a letter that has already been correctly guessed, the player loses a guess; if the player guesses a letter that is part of the secret word and has not been guessed yet, all occurrences of this letter are displayed in the correct position of the secret word, and the player does not lose a guess. The game ends when the player either has guessed all letters in the word correctly (win) or runs out of guesses (lose).

Learning Outcomes

- Practice basic programming skills
- Discover new language features[1]
- Practice interacting with the terminal window, file manager, and WingIDE
- Learn how to organize your software according to functions

Detailed Explanation and Things To Do

- Study the following screenshots to understand how the game Word Puzzle is supposed to look like and work
  - Word puzzle success 1
  - Word puzzle failure 1
  - Word puzzle success 2
  - Word puzzle failure 2
- Design and write code for a program that implements Word Puzzle as shown in the above screenshots. The program must read the instructions from a file (given below in the Resources section). The word to be guessed should be chosen randomly from a list of words maintained by the program. If the list is replaced by another list with a different number of words, the program should continue to run properly.
- Your code must implement the major logical tasks, such as displaying instructions, displaying the current puzzle state, running the main game loop, updating the puzzle state after a guess, and displaying results as ***user-defined functions*****.**

[1] To create this game, you may need to use a few programming language features that have not been used yet in class. Discovering new language features and how to use them is an integral part of problem-solving in computing science and an essential skill that you should learn. Think about what you need to do, search the web for python3 programming examples, and/or use the Python documentation to help you find the programming constructs that you need. If you get stuck, ask your TA for help/hints about the programming constructs you need to use.

- There should not be any global identifiers in the program except for module names that are imported. All other identifiers must be initialized inside a function.
- There should not be any function call statement except for the call to the main() function in the global namespace of the program.
- Make sure you are following code quality standards outlined in the Software Quality Tests.

Resources

- To display the instructions for the game use the file wp_instructions.txt.
- For testing purposes, you may use the following list of words:
    - 'apple', 'banana', 'watermelon', 'kiwi', 'pineapple', 'mango'

Submission Information

- You are required to submit the solution by the due date provided. For submission purposes, the file with your code should be named:

  `word_puzzle.py`