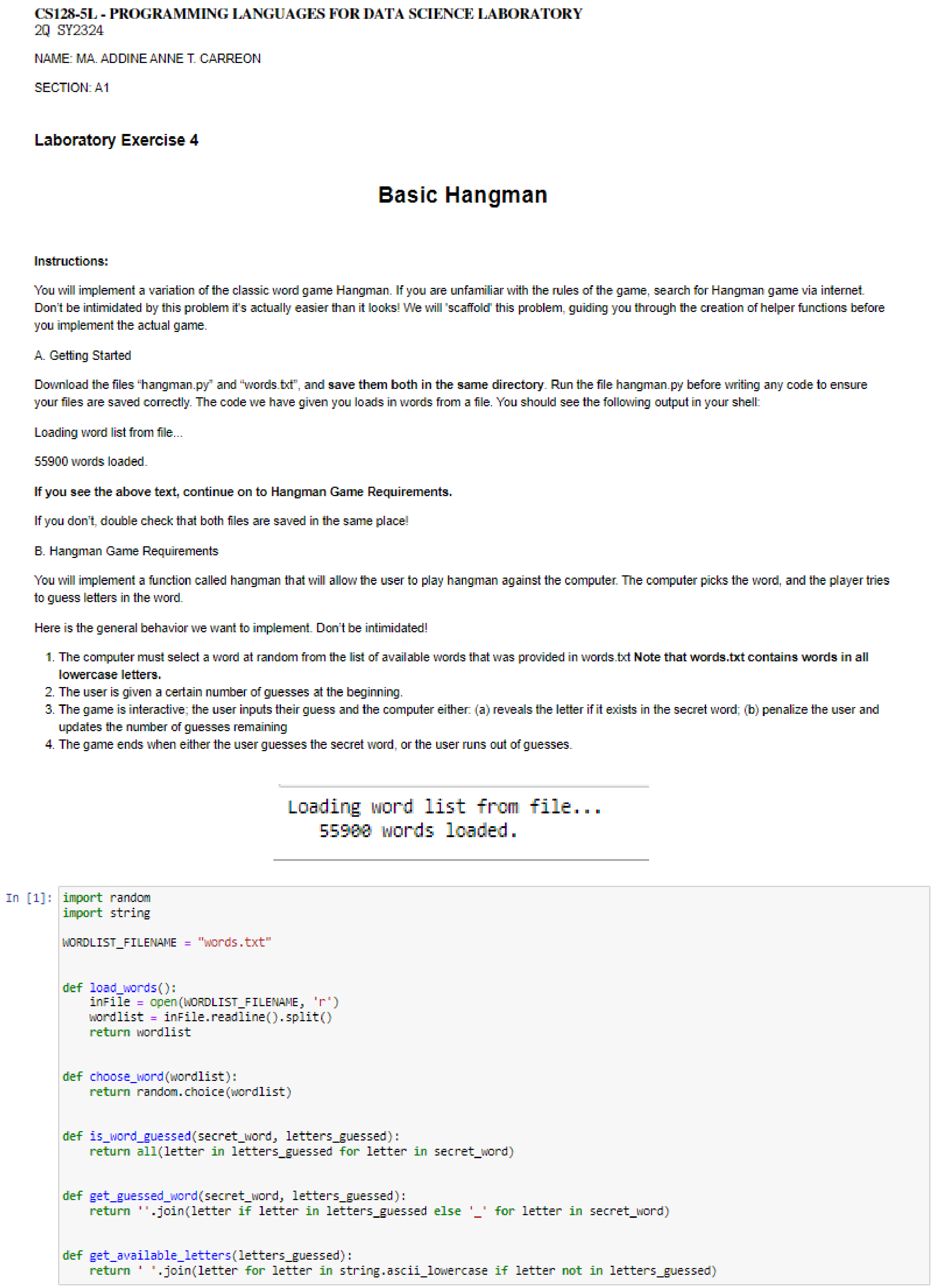
**Carreon, Ma. Addine Anne T.** **01/14/2024**
**CS128-5L – A1**



**CS128-5L - PROGRAMMING LANGUAGES FOR DATA SCIENCE LABORATORY**
2Q SY2324

NAME: MA. ADDINE ANNE T. CARREON

SECTION: A1

### Laboratory Exercise 4

## Basic Hangman

**Instructions:**

You will implement a variation of the classic word game Hangman. If you are unfamiliar with the rules of the game, search for Hangman game via internet. Don't be intimidated by this problem it's actually easier than it looks! We will 'scaffold' this problem, guiding you through the creation of helper functions before you implement the actual game.

A. Getting Started

Download the files "hangman.py" and "words.txt", and **save them both in the same directory**. Run the file hangman.py before writing any code to ensure your files are saved correctly. The code we have given you loads in words from a file. You should see the following output in your shell:

Loading word list from file...

55900 words loaded.

**If you see the above text, continue on to Hangman Game Requirements.**

If you don't, double check that both files are saved in the same place!

B. Hangman Game Requirements

You will implement a function called hangman that will allow the user to play hangman against the computer. The computer picks the word, and the player tries to guess letters in the word.

Here is the general behavior we want to implement. Don't be intimidated!

1. The computer must select a word at random from the list of available words that was provided in words.txt **Note that words.txt contains words in all lowercase letters.**
2. The user is given a certain number of guesses at the beginning.
3. The game is interactive; the user inputs their guess and the computer either: (a) reveals the letter if it exists in the secret word; (b) penalize the user and updates the number of guesses remaining
4. The game ends when either the user guesses the secret word, or the user runs out of guesses.

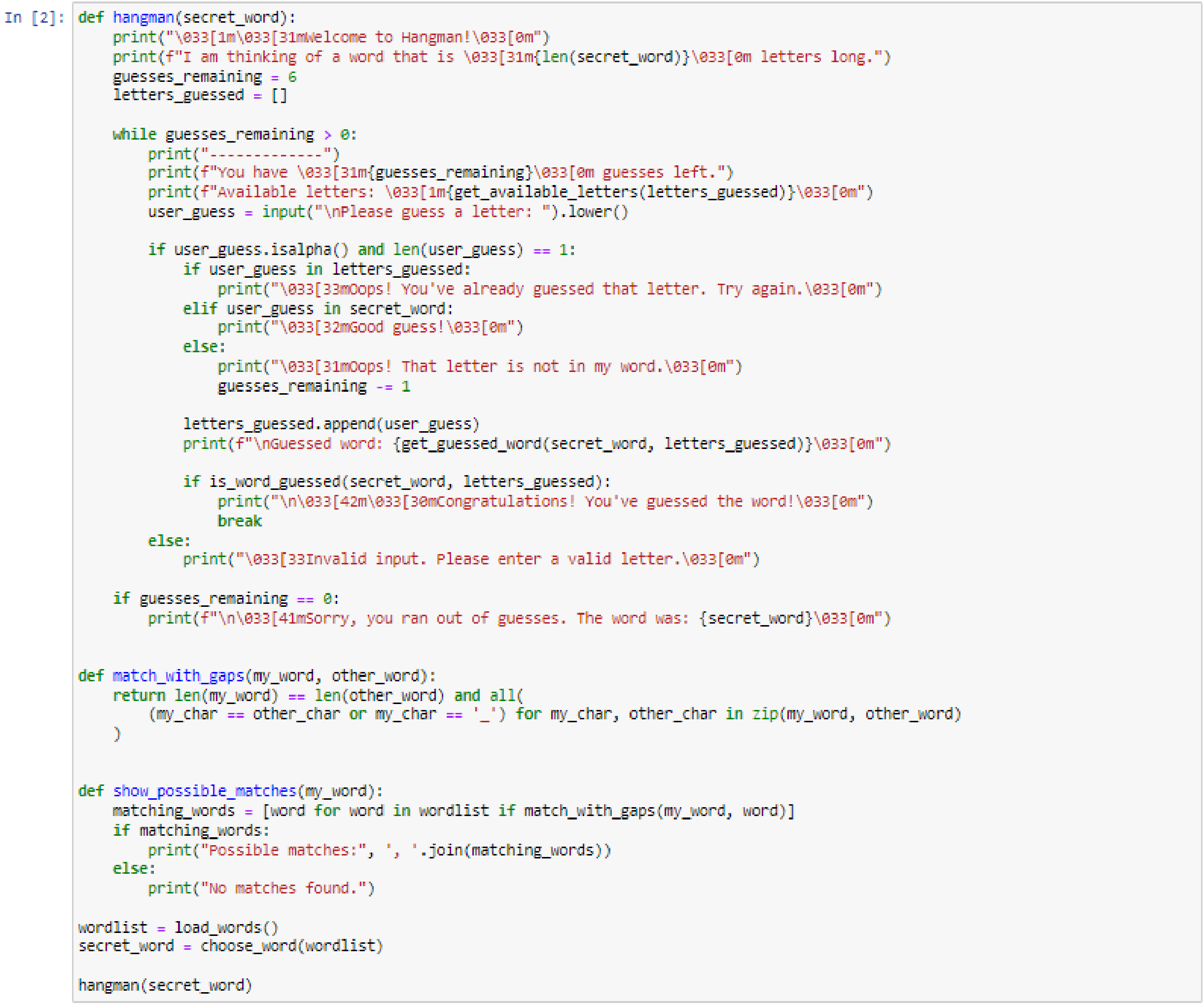
```
Loading word list from file...
    55900 words loaded.
```

In [1]:

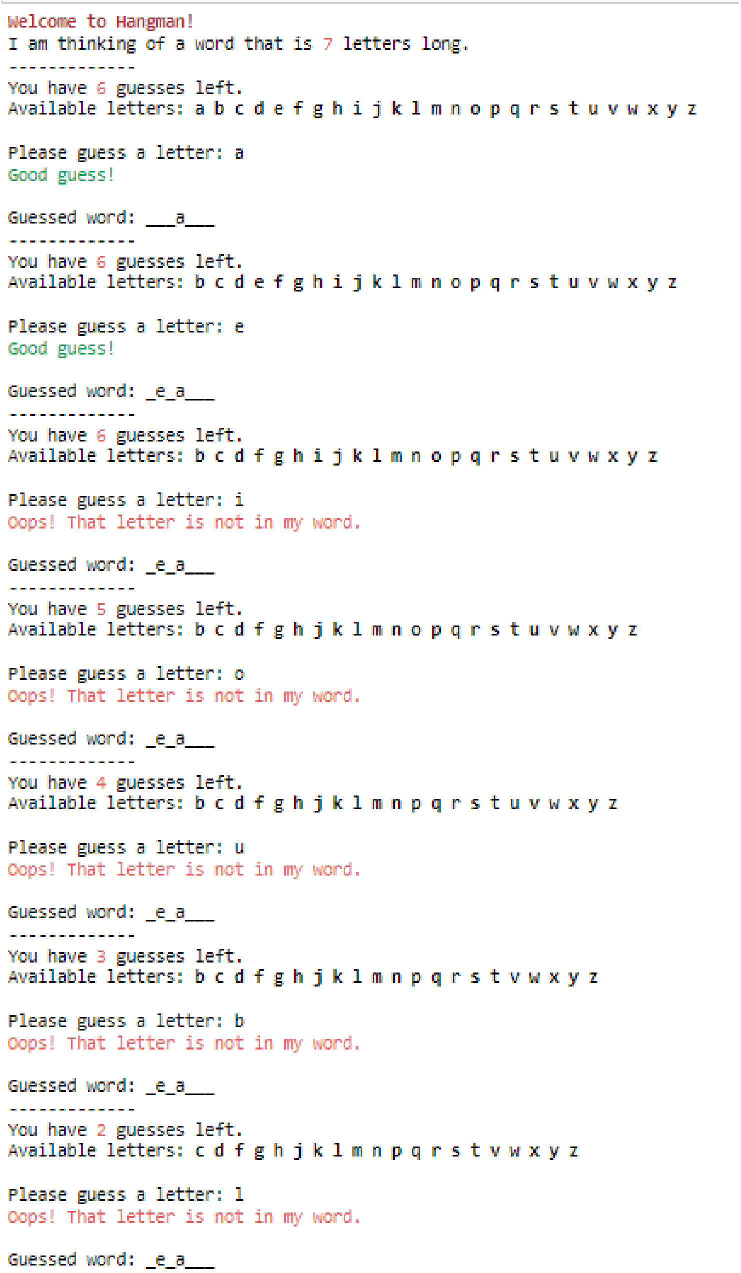
```python
import random
import string

WORDLIST_FILENAME = "words.txt"


def load_words():
    inFile = open(WORDLIST_FILENAME, 'r')
    wordlist = inFile.readline().split()
    return wordlist


def choose_word(wordlist):
    return random.choice(wordlist)


def is_word_guessed(secret_word, letters_guessed):
    return all(letter in letters_guessed for letter in secret_word)


def get_guessed_word(secret_word, letters_guessed):
    return ''.join(letter if letter in letters_guessed else '_' for letter in secret_word)


def get_available_letters(letters_guessed):
    return ' '.join(letter for letter in string.ascii_lowercase if letter not in letters_guessed)
```

**Carreon, Ma. Addine Anne T.** **01/14/2024**
**CS128-5L – A1**

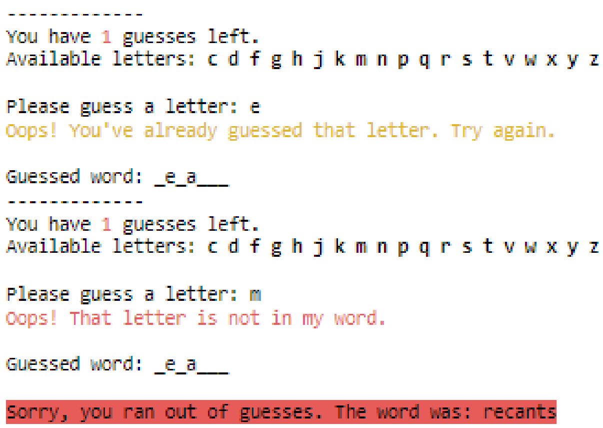
## *hangman(secret_word)*



```
In [2]: def hangman(secret_word):
            print("\033[1m\033[31mWelcome to Hangman!\033[0m")
            print(f"I am thinking of a word that is \033[31m{len(secret_word)}\033[0m letters long.")
            guesses_remaining = 6
            letters_guessed = []

            while guesses_remaining > 0:
                print("-------------")
                print(f"You have \033[31m{guesses_remaining}\033[0m guesses left.")
                print(f"Available letters: \033[1m{get_available_letters(letters_guessed)}\033[0m")
                user_guess = input("\nPlease guess a letter: ").lower()

                if user_guess.isalpha() and len(user_guess) == 1:
                    if user_guess in letters_guessed:
                        print("\033[33mOops! You've already guessed that letter. Try again.\033[0m")
                    elif user_guess in secret_word:
                        print("\033[32mGood guess!\033[0m")
                    else:
                        print("\033[31mOops! That letter is not in my word.\033[0m")
                        guesses_remaining -= 1

                    letters_guessed.append(user_guess)
                    print(f"\nGuessed word: {get_guessed_word(secret_word, letters_guessed)}\033[0m")

                    if is_word_guessed(secret_word, letters_guessed):
                        print("\n\033[42m\033[30mCongratulations! You've guessed the word!\033[0m")
                        break
                else:
                    print("\033[33Invalid input. Please enter a valid letter.\033[0m")

            if guesses_remaining == 0:
                print(f"\n\033[41mSorry, you ran out of guesses. The word was: {secret_word}\033[0m")


        def match_with_gaps(my_word, other_word):
            return len(my_word) == len(other_word) and all(
                (my_char == other_char or my_char == '_') for my_char, other_char in zip(my_word, other_word)
            )


        def show_possible_matches(my_word):
            matching_words = [word for word in wordlist if match_with_gaps(my_word, word)]
            if matching_words:
                print("Possible matches:", ', '.join(matching_words))
            else:
                print("No matches found.")

        wordlist = load_words()
        secret_word = choose_word(wordlist)

        hangman(secret_word)
```

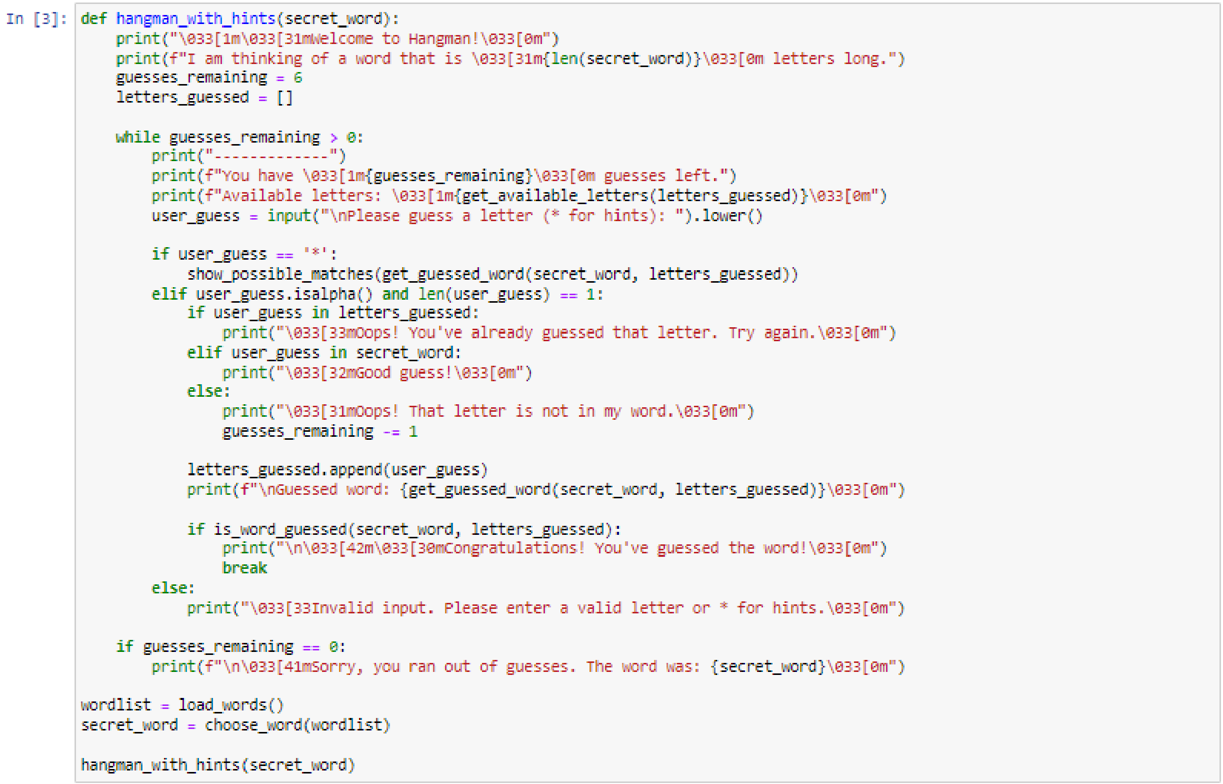


```
Welcome to Hangman!
I am thinking of a word that is 7 letters long.
-------------
You have 6 guesses left.
Available letters: a b c d e f g h i j k l m n o p q r s t u v w x y z

Please guess a letter: a
Good guess!

Guessed word: ___a___
-------------
You have 6 guesses left.
Available letters: b c d e f g h i j k l m n o p q r s t u v w x y z

Please guess a letter: e
Good guess!

Guessed word: _e_a___
-------------
You have 6 guesses left.
Available letters: b c d f g h i j k l m n o p q r s t u v w x y z

Please guess a letter: i
Oops! That letter is not in my word.

Guessed word: _e_a___
-------------
You have 5 guesses left.
Available letters: b c d f g h j k l m n o p q r s t u v w x y z

Please guess a letter: o
Oops! That letter is not in my word.

Guessed word: _e_a___
-------------
You have 4 guesses left.
Available letters: b c d f g h j k l m n p q r s t u v w x y z

Please guess a letter: u
Oops! That letter is not in my word.

Guessed word: _e_a___
-------------
You have 3 guesses left.
Available letters: b c d f g h j k l m n p q r s t v w x y z

Please guess a letter: b
Oops! That letter is not in my word.

Guessed word: _e_a___
-------------
You have 2 guesses left.
Available letters: c d f g h j k l m n p q r s t v w x y z

Please guess a letter: l
Oops! That letter is not in my word.

Guessed word: _e_a___
```



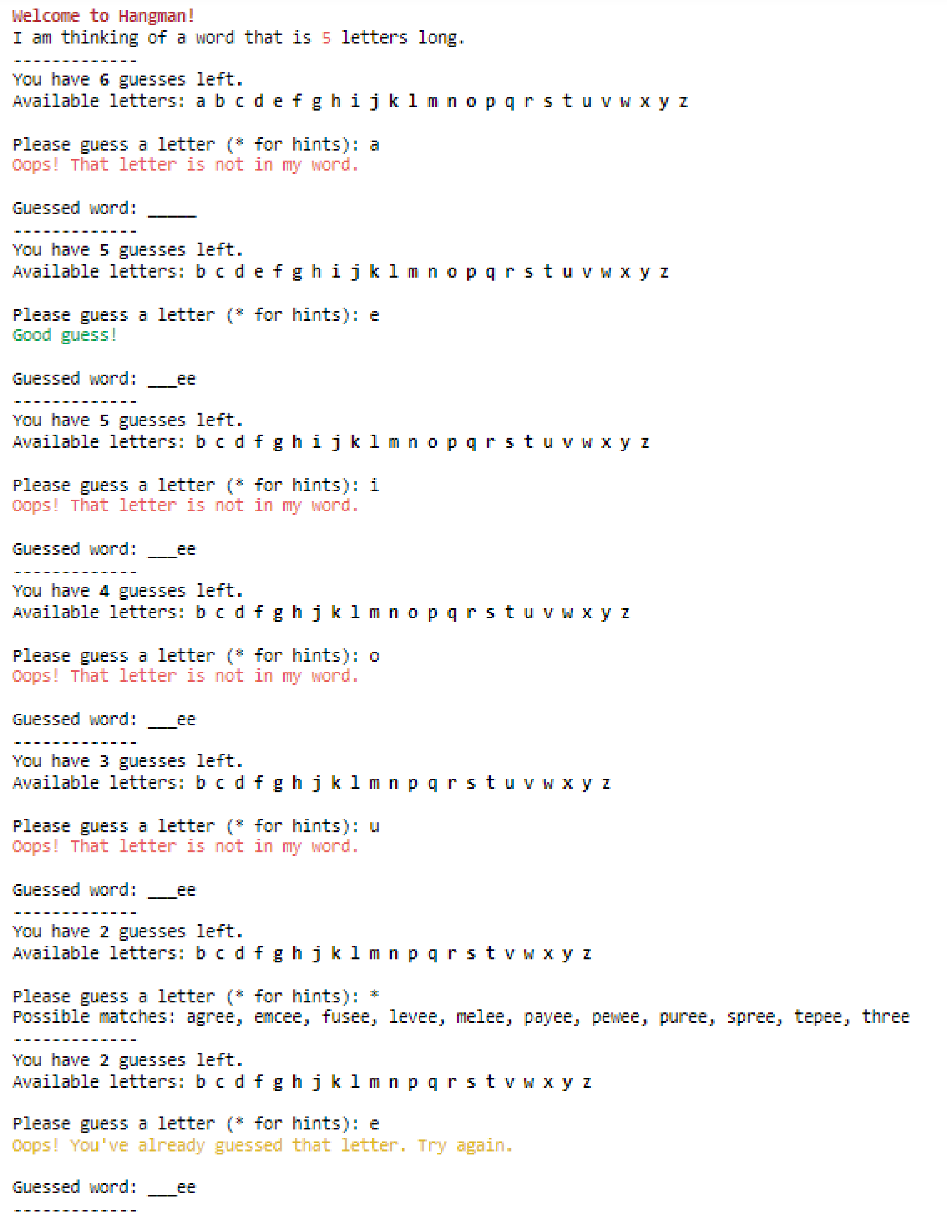
```
-------------
You have 1 guesses left.
Available letters: c d f g h j k m n p q r s t v w x y z

Please guess a letter: e
Oops! You've already guessed that letter. Try again.

Guessed word: _e_a___
-------------
You have 1 guesses left.
Available letters: c d f g h j k m n p q r s t v w x y z

Please guess a letter: m
Oops! That letter is not in my word.

Guessed word: _e_a___

Sorry, you ran out of guesses. The word was: recants
```

## *hangman_with_hints(secret_word)*

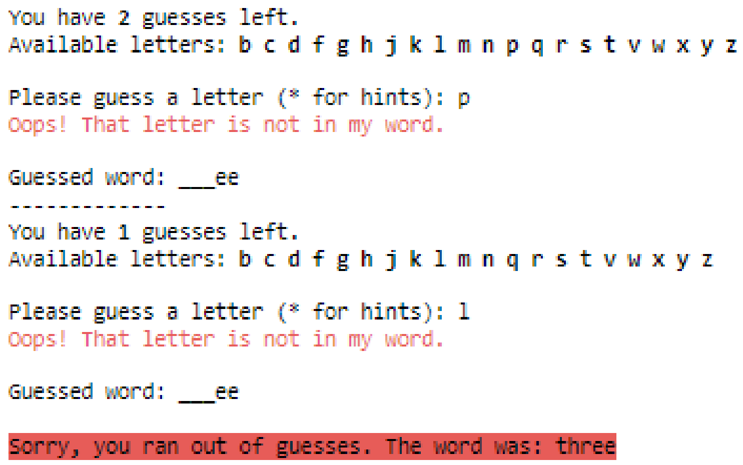


In [3]:

```
def hangman_with_hints(secret_word):
    print("\033[1m\033[31mWelcome to Hangman!\033[0m")
    print(f"I am thinking of a word that is \033[31m{len(secret_word)}\033[0m letters long.")
    guesses_remaining = 6
    letters_guessed = []

    while guesses_remaining > 0:
        print("-------------")
        print(f"You have \033[1m{guesses_remaining}\033[0m guesses left.")
        print(f"Available letters: \033[1m{get_available_letters(letters_guessed)}\033[0m")
        user_guess = input("\nPlease guess a letter (* for hints): ").lower()

        if user_guess == '*':
            show_possible_matches(get_guessed_word(secret_word, letters_guessed))
        elif user_guess.isalpha() and len(user_guess) == 1:
            if user_guess in letters_guessed:
                print("\033[33mOops! You've already guessed that letter. Try again.\033[0m")
            elif user_guess in secret_word:
                print("\033[32mGood guess!\033[0m")
            else:
                print("\033[31mOops! That letter is not in my word.\033[0m")
                guesses_remaining -= 1

            letters_guessed.append(user_guess)
            print(f"\nGuessed word: {get_guessed_word(secret_word, letters_guessed)}\033[0m")

            if is_word_guessed(secret_word, letters_guessed):
                print("\n\033[42m\033[30mCongratulations! You've guessed the word!\033[0m")
                break
        else:
            print("\033[33Invalid input. Please enter a valid letter or * for hints.\033[0m")

    if guesses_remaining == 0:
        print(f"\n\033[41mSorry, you ran out of guesses. The word was: {secret_word}\033[0m")

wordlist = load_words()
secret_word = choose_word(wordlist)

hangman_with_hints(secret_word)
```



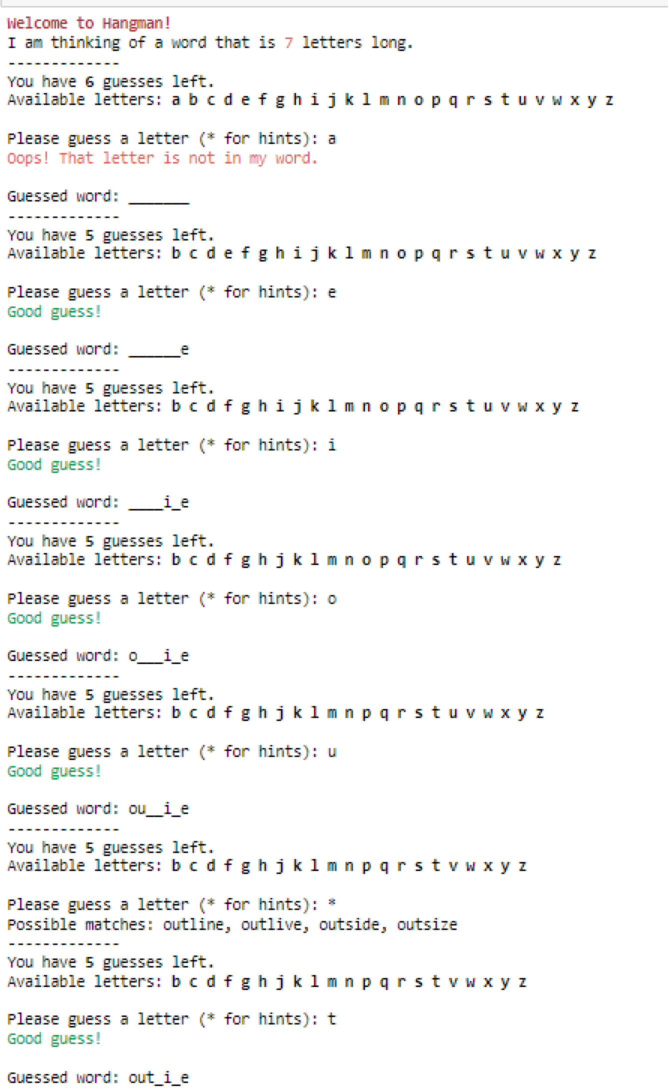
```
Welcome to Hangman!
I am thinking of a word that is 5 letters long.
-------------
You have 6 guesses left.
Available letters: a b c d e f g h i j k l m n o p q r s t u v w x y z

Please guess a letter (* for hints): a
Oops! That letter is not in my word.

Guessed word: _____
-------------
You have 5 guesses left.
Available letters: b c d e f g h i j k l m n o p q r s t u v w x y z

Please guess a letter (* for hints): e
Good guess!

Guessed word: ___ee
-------------
You have 5 guesses left.
Available letters: b c d f g h i j k l m n o p q r s t u v w x y z

Please guess a letter (* for hints): i
Oops! That letter is not in my word.

Guessed word: ___ee
-------------
You have 4 guesses left.
Available letters: b c d f g h j k l m n o p q r s t u v w x y z

Please guess a letter (* for hints): o
Oops! That letter is not in my word.

Guessed word: ___ee
-------------
You have 3 guesses left.
Available letters: b c d f g h j k l m n p q r s t u v w x y z

Please guess a letter (* for hints): u
Oops! That letter is not in my word.

Guessed word: ___ee
-------------
You have 2 guesses left.
Available letters: b c d f g h j k l m n p q r s t v w x y z

Please guess a letter (* for hints): *
Possible matches: agree, emcee, fusee, levee, melee, payee, pewee, puree, spree, tepee, three
-------------
You have 2 guesses left.
Available letters: b c d f g h j k l m n p q r s t v w x y z

Please guess a letter (* for hints): e
Oops! You've already guessed that letter. Try again.

Guessed word: ___ee
-------------
```



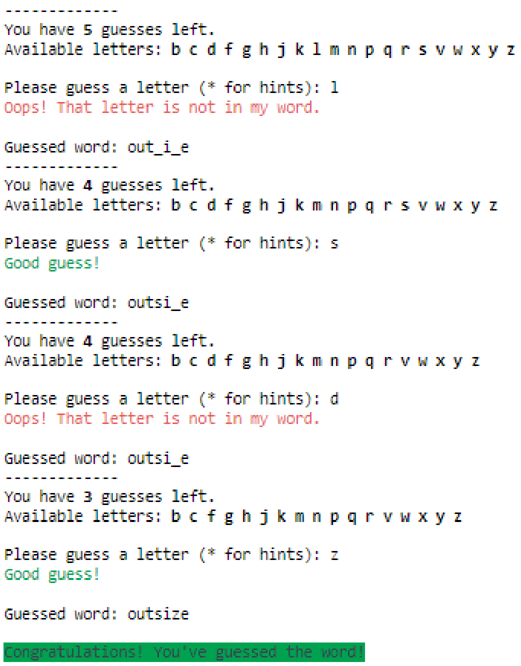
```
You have 2 guesses left.
Available letters: b c d f g h j k l m n p q r s t v w x y z

Please guess a letter (* for hints): p
Oops! That letter is not in my word.

Guessed word: ___ee
-------------
You have 1 guesses left.
Available letters: b c d f g h j k l m n q r s t v w x y z

Please guess a letter (* for hints): l
Oops! That letter is not in my word.

Guessed word: ___ee

Sorry, you ran out of guesses. The word was: three
```

**Carreon, Ma. Addine Anne T.** **01/14/2024**

**CS128-5L – A1**



```
Welcome to Hangman!
I am thinking of a word that is 7 letters long.
-------------
You have 6 guesses left.
Available letters: a b c d e f g h i j k l m n o p q r s t u v w x y z

Please guess a letter (* for hints): a
Oops! That letter is not in my word.

Guessed word: _______
-------------
You have 5 guesses left.
Available letters: b c d e f g h i j k l m n o p q r s t u v w x y z

Please guess a letter (* for hints): e
Good guess!

Guessed word: ______e
-------------
You have 5 guesses left.
Available letters: b c d f g h i j k l m n o p q r s t u v w x y z

Please guess a letter (* for hints): i
Good guess!

Guessed word: ____i_e
-------------
You have 5 guesses left.
Available letters: b c d f g h j k l m n o p q r s t u v w x y z

Please guess a letter (* for hints): o
Good guess!

Guessed word: o___i_e
-------------
You have 5 guesses left.
Available letters: b c d f g h j k l m n p q r s t u v w x y z

Please guess a letter (* for hints): u
Good guess!

Guessed word: ou__i_e
-------------
You have 5 guesses left.
Available letters: b c d f g h j k l m n p q r s t v w x y z

Please guess a letter (* for hints): *
Possible matches: outline, outlive, outside, outsize
-------------
You have 5 guesses left.
Available letters: b c d f g h j k l m n p q r s t v w x y z

Please guess a letter (* for hints): t
Good guess!

Guessed word: out_i_e
```



```
-------------
You have 5 guesses left.
Available letters: b c d f g h j k l m n p q r s v w x y z

Please guess a letter (* for hints): l
Oops! That letter is not in my word.

Guessed word: out_i_e
-------------
You have 4 guesses left.
Available letters: b c d f g h j k m n p q r s v w x y z

Please guess a letter (* for hints): s
Good guess!

Guessed word: outsi_e
-------------
You have 4 guesses left.
Available letters: b c d f g h j k m n p q r v w x y z

Please guess a letter (* for hints): d
Oops! That letter is not in my word.

Guessed word: outsi_e
-------------
You have 3 guesses left.
Available letters: b c f g h j k m n p q r v w x y z

Please guess a letter (* for hints): z
Good guess!

Guessed word: outsize

Congratulations! You've guessed the word!
```