ECE326 Lab 3 Report

Design a new utspell program similar to the unix spell program.
My utspell program allows users to pass in input through (1) a file/files or (2) the command line, similar to the unix spell program. To pass in a file/files, this can be done by running `python3 shaocass_lab3_code.py -f file.py` for a single file, or running `python3 shaocass_lab3_code.py -f file1.py file2.py file3.py` for multiple files. To pass in a word or a sentence directly in the command line, this can be done by running `python3 shaocass_lab3_code.py -c word` for a single word, or `python3 shaocass_lab3_code.py -c this is a sentence to check for misspelling` for a sentence. The program iterates through the input words (from the file(s) or command line) and checks to see if they are present in the list of words from /usr/share/dict/words. If a word is not found in /usr/share/dict/words, the program adds that word to a set of misspelled words. A set is used to ensure that the misspelled words that are stored are unique, similar to the unix program. The program ends with printing the unique misspelled words each on their own line in the terminal.

Analyze utspell and also collate possible future enhancements.
The utspell program checks the dictionary for each word that is passed in, in the input. An enhancement to this can be to sort the input words and remove duplicates. This would ensure that we only check the dictionary for unique input words, and we do not check the dictionary multiple times for the same word/spelling. There is also room for qualitative future enhancements. Currently, the utspell program outputs information in the same way as the unix spell program–it prints the unique misspelled words each on their own line. Some enhancements here can be to indicate the locations (i.e., indices) of the misspelled word and the number of times that the misspelling appears. Another enhancement can be to have the option to add more words to the dictionary (e.g., names), to prevent those from being considered as misspelled.

Write a small note on how to make it a spell checker for French.
The list of words found at /usr/share/dict/words gives the English dictionary words. To make a spell checker for French, I need to replace /usr/share/dict/words with one that gives the French dictionary words. The other logic of the program remains the same.

Code:

```python
# Lab 3

# Design a new utspell program similar to unix spell program.
# Use /usr/share/dict/words as the English dictionary.

import sys
import string

def utspell():

    english_dict = open('/usr/share/dict/words').read().split('\n')

    misspelled = set() # Initialize a set to hold the unique misspelled words.

    translate_punc = str.maketrans(string.punctuation, ' '*len(string.punctuation)) # Translator for replacing punctuation with spaces.

    if len(sys.argv) <= 1:
        print("No words supplied.")
        return

    if sys.argv[1] == "-f": # Read from file(s).
        i = 2
        while i < len(sys.argv):
            file_data = open(sys.argv[i], 'r').read().translate(translate_punc).split() # Replace punctuation.
            for word in file_data:
                if word not in english_dict:
                    misspelled.add(word)
            i += 1
    elif sys.argv[1] == "-c": # Read from command line.
        data = ' '.join(sys.argv[2:])
        data = data.translate(translate_punc) # Replace punctuation.
        data = data.split()
        for word in data:
            if word not in english_dict:
                misspelled.add(word)

    # Print each element in the set of misspelled words on its own line.
    for word in misspelled:
        print(word)

utspell()
```