Homework 4: Loops, I/O, & (some) Strings

L435/L555

Due Thursday, October 6

1. This program from a hypothetical student is supposed to read in 50 words, put them into a list, sort them, and then print every one with a count of how many more words are to be printed. Unfortunately, there are (depending on how you count them) 5 mistakes in the program. Find them and correct them.

```
words = {}
for x < 50:
    words.append(input('next word: '))
words.sorted()
for x in range(1,50):
    print(words[x], 'words to do:', 50-x)
```

2. A Fibonacci number is calculated by taking the sum of the previous two numbers, starting from 0 and 1. So, the first ten Fibonacci numbers are: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34.

   (a) Write a program that prompts a user for a number (n) and uses a `while` loop to calculate the n^{th} Fibonacci number. Remember that $f_i = f_{i-2} + f_{i-1}$ and so you'll need variables to account for this.

   (b) Write an alternative version of the program which uses a `for` loop instead.

3. Write a program that reads in 100 words from the terminal (i.e., as part of a pipeline) and creates a frequency lexicon. Store the words in a list and the frequency counts in another list so that the frequency count for a specific word has the same position as the word. Make sure that if a word is already in the list, it should not be added, but only the frequency count must be increased. Finally, output the list.

   To test your program use the file wordlist100.txt from canvas under Files/data.

4. (a) Write a program that reads in a sentence and returns all the words that do not contain the letter 'e'. (A *word* here is defined as any string occuring between whitespace.)

   (b) Write a program that takes a sentence and a string of forbidden characters and returns a message to say whether any of these characters are in the sentence.

5. L555 only (extra credit for L435): Taking your Mini-MASC data from Assignment 2, write a Python program which inputs a `.ptb` file of your choice and counts up the number of lines containing *NP*.