CPSC 313: Distributed and Cloud Computing
Spring 2022

Lab 1: Count words in files

Summary

You will create a single python code file named "count_words.py" that takes a list of files as input, and count all the words in the list of files, printing out the list of words with their counts.

You should have the following constants & functions:

```
# stop words are words that don't count – conjunctions, extensions, prepositions, etc. Use this list to check each word you find, and if it's in this list you ignore it.
STOP_WORDS =
['a', 'an', 'and', 'are', 'as', 'be', 'by', 'for', 'if', 'in', 'is', 'it', 'of', 'or', 'py', 'rst', 'that', 'the', 'to', 'with',]


def count_words(file_name):   # take a file and output the word list with the counts for that file as a tuple (or list).  This may look like:  [['foo', 22], ['bar', 13], ...]


def main():  # main function: get the files from input (or *args). Iterate through the files counting words in each file, then combine results from each

if __name__ == '__main__':
   main()
```

Requirements

- You should have a test file that tests both your count_words function and your main loop.
- I suggest you use pytest as your test framework, though any python test framework is fine. Look up pytest to see how to use it, though it's simple.
    - Install pytest – in your terminal within vscode or in a powershell window:
        - >> pip install pytest
    - Make sure in your test file each function (or class) you use has "test" in the name. As a convention, I tend to start each function with "test". So if I'm testing my word_count function, I'd have a "test_word_count" function in my test file.
    - To run the tests, just use the command "pytest" in your powershell or terminal

- You must log relevant information from your working code into a log file: "count_words.log"
    - Use the Python “logging” library. To see how to use that library, search for instructions.
- You must compute the total time it takes to count all the words in all the files, and output that clearly to the log file.
- You must follow the style guide (document is in Teams) with the following exceptions:
    - naming: Single letter variable or function names are NOT allowed.
        - All names must be meaningful and descriptive. Instead of "for x in list:" you may use "for loop_control in list" or something more meaningful.
    - You may use tabs instead of spaces, and your indentation is encouraged to be tab or 4 spaces instead of 2 spaces.
- You can find test files by searching for “random word test file”
- For this assignment you are not required to test large files, but for the next iteration of this assignment you will. If you want to get a jump on that, you may want to write code to generate a large text file. Some hints:
    - Take the words you find in the smaller test files, and make that the word corpus by reading them all into a list of source words.
    - Take a random word by generating a random index into the list
    - Write that word to a new file and continue until you reach the size you want

Submission

- Your python file (count_words.py)
- Your test file (test_CW.py)
- A log file that shows the output of your test run and any log statements from the code