T.C.
DOKUZ EYLUL UNIVERSTY

FACULTY OF ENGINEERING

DEPARTMENT OF COMPUTER ENGINEERING

2021 - 2022
FALL SEMESTER

CME 1203
INTRODUCTION TO COMPUTER ENGINEERING

ASSIGNMENT 1
WORD ORDER FINDER

DUE DATE
23:55 – 26.12.2021

In this assignment, you are asked to create a Python file that contains two different functions for text processing. You can use the books available from Gutenberg Project (https://dev.gutenberg.org/browse/scores/top) for your program.

The first of these functions will process the book(s) you downloaded and output the frequency of word order (in descending order) with selected sizes for one book. The second function will do the same operation for two different books and output the sum of the frequency of word order (in descending order) with selected sizes. The structure and explanation of these functions are given below.

Word_Order_Frequency_One_Book (Book, Word_Order, File_Output)
Word_Order_Frequency_Two_Books (Book_1, Book_2, Word_Order, File_Output)

Book, Book_1 and Book_2: Contains the name of text file(s) that your book(s) is/are stored in. It should be in the same directory as your program. Your function should handle exceptions and errors related to this input if it is incorrect.

Word_Order: The number of words that will be ordered and analyzed by your function. It must be a positive integer and you should consider only 1 (a single word) and 2 (adjacent two words). If it is given as 1, it will output single word frequency of the book in contrast to a specific ordered sequence of word frequency.

File_Output: The name of the text file the results of this program will be saved. If the file does not exist, you should create it and if it exists, should you override the previous contents. It must be a string that contains a file name. Your program should not output any results to the console, only to the file identified with this variable.

The basic algorithm of these functions are given below.

Firstly, read the given file or files.

Tokenize the words (separate them according to empty spaces)

Remove the stop words
(You can use the given file “stop_words_english.txt” for this purpose)

Remove punctuation symbols and other incorrect elements.
(You can use this site https://www.ascii-code.com/, Do not consider other punctuation marks that are not an element of ASCII or Extended ASCII character groups)

Generate word order sequences and calculate their frequency.

Print out the results to the given text file.

The main difference between these functions are in word sequence frequency calculation and result print out stage. For “Word_Order_Frequency_One_Book” function, the output should look like the following.

```
| WORD      | WORD     |
| ORDER     | ORDER    |
| FREQUENCY | SEQUENCE |
------------------------
 <freq_bk_1>|<word_list>
```

For “Word_Order_Frequency_Two_Books” function, the output should look like the following.

```
| TOTAL     | BOOK 1    | BOOK 2    | WORD     |
| ORDER     | ORDER     | ORDER     | ORDER    |
| FREQUENCY | FREQUENCY | FREQUENCY | SEQUENCE |
------------------------------------------------
<freq_total>|<freq_bk_1>|<freq_bk_2>|<word_list>
```

<freq_total>, <freq_bk_1> and <freq_bk_2>: The frequency values of word order sequences.. You should order them correctly from right to left as numbers. You should also increase the width of columns if the numbers are larger than given default size.

<word_list>: The list of words the represents the word order sequence. You should print out the word order as a single string (e.g. “Project Gutenberg”).

You should consider only English language books from Gutenberg Project (https://dev.gutenberg.org/browse/scores/top) as your input for this project because the evaluation of your homework will be conducted with randomly selected English books from this source.

When you finished writing your Python file with these functions, you should try to execute it from another Python file with a similar code to given code below.

```
import importlib

Assignment_1 = importlib.import_module(“2021510123_fatih_dicle.py”)

Assignment_1.Word_Order_Frequency_One_Book(“book_1.txt”, 2, “result_1.txt”)

Assignment_1.Word_Order_Frequency_Two_Books(“book_1.txt”, “book_2.txt”, 5, “result_2.txt”)
```

Because this is the method that will be used to execute and grade your assignments. For this reason, try to execute your assignments with the given method above with different inputs to check if it works correctly.

You should also focus on writing your code, your comments and your variables with correct English grammar, vocabulary and using the correct naming practices for your variables to make them, and your code in general, understandable to other persons.

For this assignment you will be given three text files as examples and to make your programming easier.

First is “stop_words_english.txt” that contains the stop words of English language that you are required to remove from your text. It was downloaded from this website (https://countwordsfree.com/stopwords).

Second is “book_1.txt” that contains “Moby Dick; Or, The Whale by Herman Melville” book that has been downloaded from (https://dev.gutenberg.org/ebooks/2701).

Third is “book_2.txt” that contains “A Tale of Two Cities by Charles Dickens” book that has been downloaded from (https://dev.gutenberg.org/ebooks/98).

Even though these books are given to you as examples, your code should work correctly for other books in Gutenberg Project website. Your code will be graded using other randomly selected books from this website (https://dev.gutenberg.org/).

Lastly, you should try to lowercase all words to prevent your program to consider them different from one another. You should also remove or delete punctuation marks present in the text (e.g. ?, !, :, etc.), to focus on words alone. You should consider only punctuation marks that are present in ASCII and Extended ASCII table in the following website (https://www.ascii-code.com/).

UPLOAD REQUIREMENTS:

You are required to only upload a single Python file that contains the requested functions above. Please do not upload any text files containing books you have used to test your program or secondary Python file you used to execute your main Python file. Your assignments will be evaluated using either Spyder IDE or Python console commands. Therefore, please make sure your program executes correctly with both methods.

The format of the file you are required to upload are given below with an explanation and an example. Please make sure you use lowercase and English only characters to prevent problems with library import process of Python language.

(STUDENT_NUMBER)_(STUDENT_NAME).py
(Source code you have written in Python language)
Example = 2021510123_fatih_dicle.py

Late submissions and no submissions will be graded zero. You can see the basic grading table of this assignment below.

CRITERIA	GRADE
Correct naming of upload files	10
Correct English variable names and English comments.	10
Correct Code Quality and Readability	10
Correct Error and Incorrect Input Handling	10
Correct Execution and File Output	60
TOTAL GRADE	100
CHEATING OR ANY OTHER FORM OF PLAGIARISM	–∞

You are forbidden to share your source code or writing your assignment together with other students to prevent cheating and plagiarism. However, you may share and compare the result file of your assignment to check the validity of the generated results.

If you have any questions or problems regarding this lab paper, you can ask about it in our lab sessions. If you wish, you can also ask it in class forums or assignment page comments.If you send an email and if your question is answered, please share this information with other students to prevent asking of the same question again and again.

GOOD LUCK TO YOU ALL!