**Group Members: Loukas Mironidis, Caitlyn Lambert**

**Big Data Capstone**

**Assignment 2**

**3/7/2017**

**OUTPUT FROM PART D:**

Question 1 -

Number of words: 376

Question 2 -

Average word length: 13

Question 3 -

Longest Word: "Imagepx-Distribution\_of\_Races\_on\_the\_Balkans\_in\_.jpg|px-Distribution\_of\_Races\_on\_the\_Balkans\_in\_.jpg"

Word Length: 102

Question 4 -

Number of Hapax words: 138 Percent: 36.70

Question 5 -

Frequent word: uploaded Number of times: 45 Percent: 11.97

Frequent word: Σεπτ Number of times: 27 Percent: 7.18

Frequent word: UserLeonariso|Leonariso Number of times: 17 Percent: 4.52

Frequent word: (flag) Number of times: 13 Percent: 3.46

Frequent word: Συζήτηση Number of times: 9 Percent: 2.39

Frequent word: UserStefanosKozanis|StefanosKozanis Number of times: 9 Percent: 2.39

Frequent word: Νοέμβ Number of times: 8 Percent: 2.13

Frequent word: UserAnk|Ank Number of times: 7 Percent: 1.86

Frequent word: Δεκέμ Number of times: 6 Percent: 1.60

Frequent word: by Number of times: 5 Percent: 1.33

Total Percent: 38.83

**Question 1 - How big is the corpus (# of words)?**

I found this by using the method called findall() from library re. This method takes in

two arguments a regular expression, and target string which was my cleaned data from part B.

My regex pattern was to find any spaces one or more times within the text. I would then use the len()

method to count the number of words within the text file.

**Question 2 - What is the average word length?**

I found the average word length by using the split() method and looping through my text file saving the string within another variable where I used the len() to find the length. After this I simply just used the sum() to divide the average by the length of the text file to find the average word length.

**Question 3 - Which is the longest word?**

I found the longest word by first creating an empty string and looping through my text file using the split() method saving the string inside another variable. I then checked for the longest word by creating an 'if' statement where if the length of the word is greater than the length of the empty string I created set str = word. After the for loop finishes I simply print the longest word and the length of the word.

**Question 4 - How many hapax words are there? How many percent of the corpus are they?**

I found this by first opening the output file from partB again and creating two unordered collections of items so that I can have no duplicates within a set. One set will have all the words from the text while the other one will only have duplicates. I then loop through multiple times and use an 'if' statement asking if the word being searched is in the set collection, add the word into duplicates and allwords set collections. I then just simply subtract allwords and duplicates set collections to find the unique number of Hapax words in the text file. Also within the print statement I used the round() method and divided the length of the unique string divided by the total number of words from Question 1 multiplied by 100 to find the percent value of how many percent of the corpus are they.

**Question 5 - What are the 10 most frequent words? How many percent of the corpus consists of these words?**

I found this by first using the split() method and creating loop to find unique words which will be saved inside an array. I then created another for loop to iterate over the unique word array I created that will also check if the word is equal to the current unique word and appends them as tuples as (count, unique). After this second for loops is done, it is followed by two methods sort() and reverse() in order to output the correct words. The finally part of this just loops through the counts array I also created to hold the tuples and output the 10 most frequent words, number of times, as well as the percent value of the current word. To find the percent of the corpus consists of these words, I added all the counts and divided by the number of words in the text file from Question 1 and multiplied by 100 to have in percent format.

I also found another way to found this question out using a method from collections import Counter as well as importing the re library to use findall() method. This will give me the same exact output as I have in part D, just much simpler and easier to read.

import re

from collections import Counter

words = re.findall('\s+', open('test').read().lower())

print(Counter(words).most\_common(10))