

**Lab Report 5**

**Submitted by: Submitted to:**

Anmol Thapa Birodh Rijal

525 (Artificial Intelligence Lecturer)

2019 ‘A’

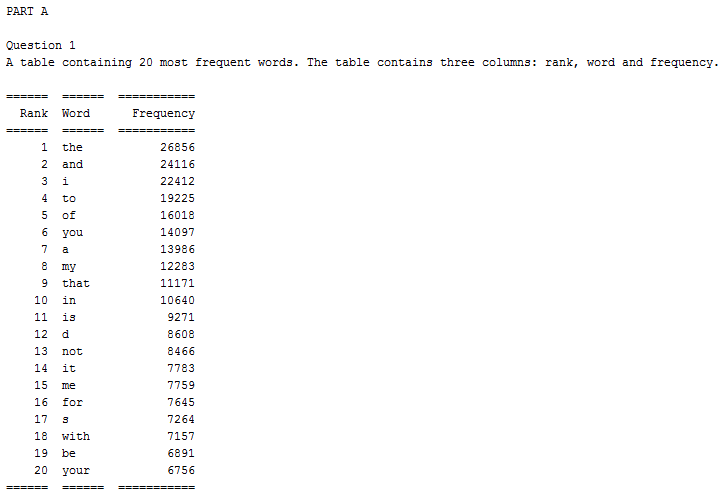
# **1.2.1 Part A**

**Your task is to read the contents of the file and produce:**

1. **A table containing 20 most frequent words. The table contains three columns: rank, word and frequency.**

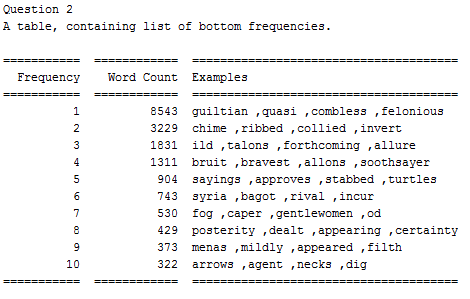
First, file read and all the words were inserted in a list. Then, I iterated through the list to create a dictionary where the key was the word and the value was the frequency of the word. Then I displayed the key-value pairs in the descending order of the values (the frequencies).

**I have used a library tabulate, so this library must be installed to run the script provided. Its use is to display outputs as included in the figures below.**



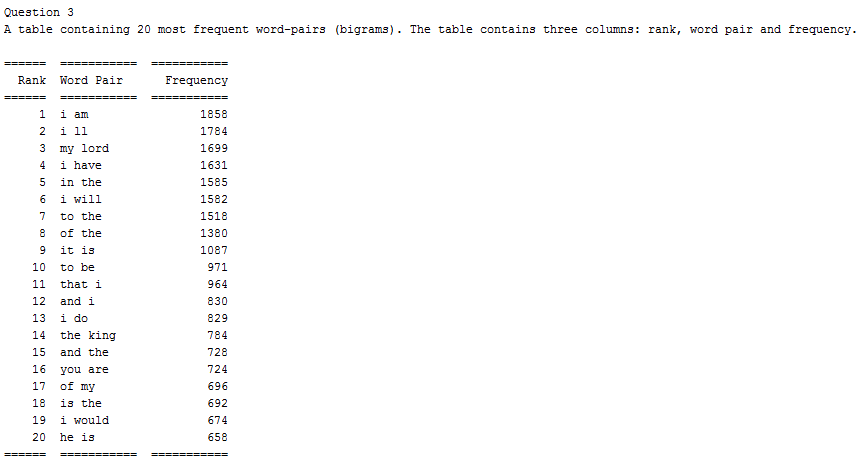
1. **A table, containing list of bottom frequencies. The table contains three columns: frequency, word count and example words. You are supposed to print word counts for frequencies 10 to 1. The rows in this table show how many words have frequency 10, 9 , 8...1 with example of some of the words.**

To, solve this question I iterated the dictionary in ascending order of values (the frequencies). The element was added to the example list. Whenever a change in frequency was detected say 1 to 2, then a new list was created which has 3 elements the rank, frequency and the example list. Then I printed the list in ascending order of rank.



1. **A table containing 20 most frequent word-pairs (bigrams). The table contains three columns: rank, word pair and frequency.**

I used the word list to create a list of bigrams, then I created a dictionary as in question 1. The key part was the bigram and the value was the frequency of the bigram. Then I iterated through the dictionary in descending order of the value (frequency bigram).



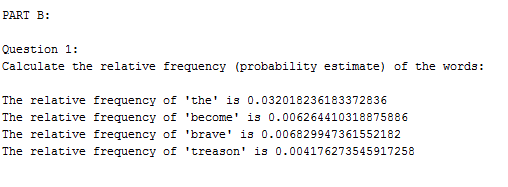
# **1.2.2 Part B**

With the frequency counts of the word at our hand we calculate some basic probability estimates.

1. **Calculate the relative frequency (probability estimate) of the words: (a) “the" (b) “become" (d) “brave" (e) “treason" [Note: P(the) = count(the) / N . Here, count(the) is the frequency of “the" and “N" is the total word count.]**

The relative frequency of a word is calculated as P(word) = count(word)/total\_no\_of\_word

The count of a word is determined from the dictionary (where the key is the word and the value id the frequency). The total\_no\_of\_words is the length of the wordlist itself.

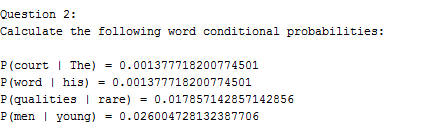


1. **Calculate the following word conditional probabilities: (a) P(court | The) (b) P(word | his) (c) P(qualities | rare) (d) P(men | young) [Read P(B | A) as “the probability with which word B follows word A". Note: P(B | A) = count(A;B) | count(A) ]**

The probability is calculated as

P(A/B) = count(a,b)/count(a)

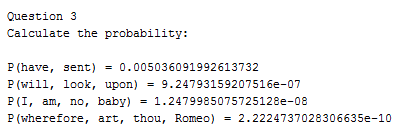
count(a,b) was extracted from the bigram dictionary and count(a) was extracted from word dictionary.



1. **Calculate the probability: (a) P(have, sent) (b) P(will, look, upon) (c) P(I, am, no, baby) (d) P(wherefore, art, thou, Romeo).**

Using Markov assumption, the probability of P(A,B,C,D) is calculated as

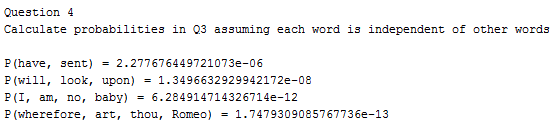
P(A,B,C,D) = P(A) \* P(B|A) \* P(C|B) \* P(D|C)



1. **Calculate probabilities in Q3 assuming each word is independent of other words (independence assumption).**

If the words are considered to be independent the probabilities are given as

P(A,B,C,D) = P(A)\*P(B)\*P(C)\*P(D), the result is shone below.



1. **Find the most probable word to follow this sequence of words: (a) I am no (b) wherefore art thou**

To solve this all the words that come after the last words (here ‘no’ and ‘thou’) were added to a list. For this the word list was used. Then, conditional probability of those words coming after the given word sequence was calculated and the element having the highest probability was used. The output is illustrated below.

