**CSE-3024 Web Mining**

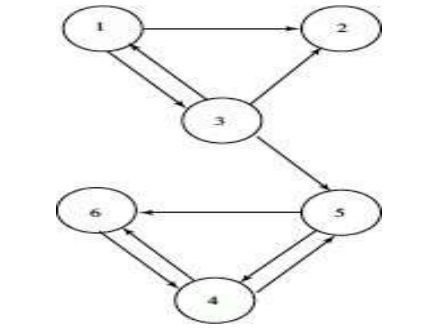
**Page Rank**

**Alokam Nikhitha**

**19BCE2555**

Question

Write a python program to find the ranks for the given graph.



Perform 7 iteration and print the final iteration value only.

**Problem statement:**

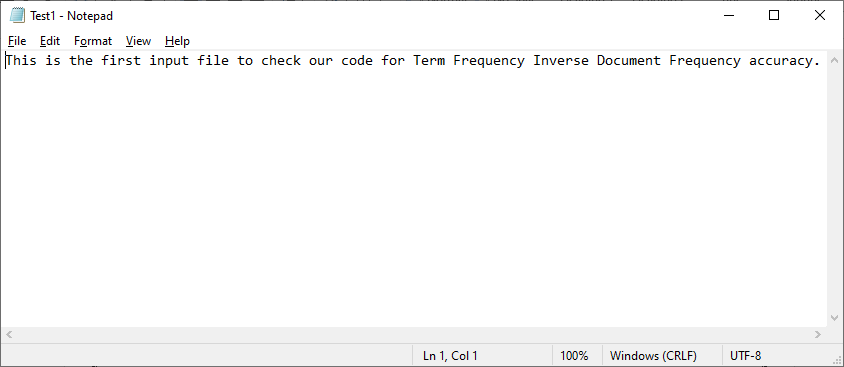
Python program to find the Page Rank of all the nodes for the given Graph after 7 iterations.

**Procedure:**

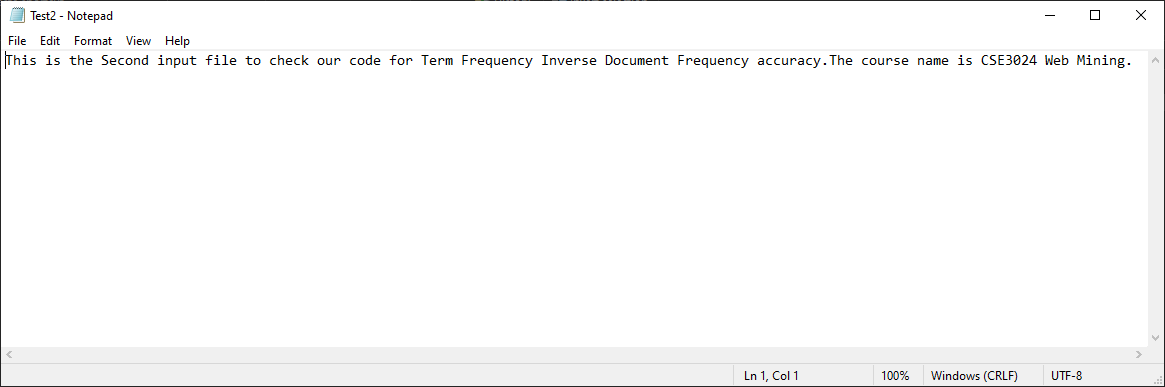
* We will Firstly import our libraries Which are required in doing the term frequency count.
* Next we will
* We will create 5 Text File inputs and read them in our workspace.
* Later, We will make the bloblist that contains all the Text File Inputs in list format. And then we will print the counts of top 3 words in every document.
* We will then calculate the cosine similarity using inbuilt cosine\_similarity matrix.
* For the above we need to create a pandas data frame of count vectors.

**Text File Taken as Input:**

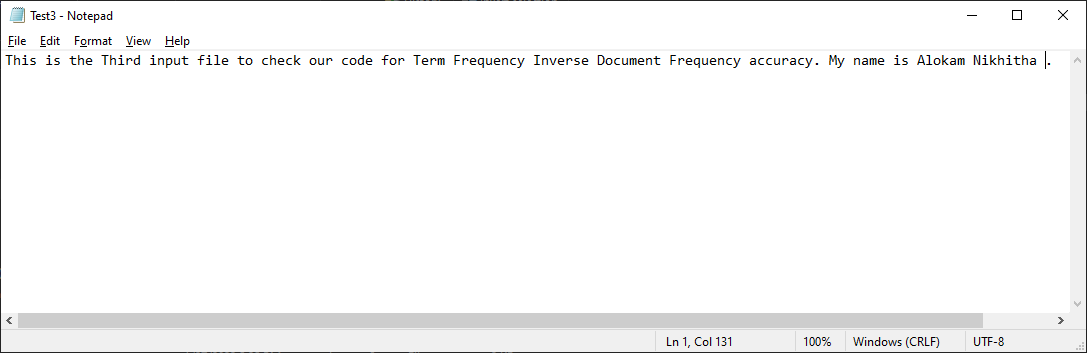
**Text File 1:**

****

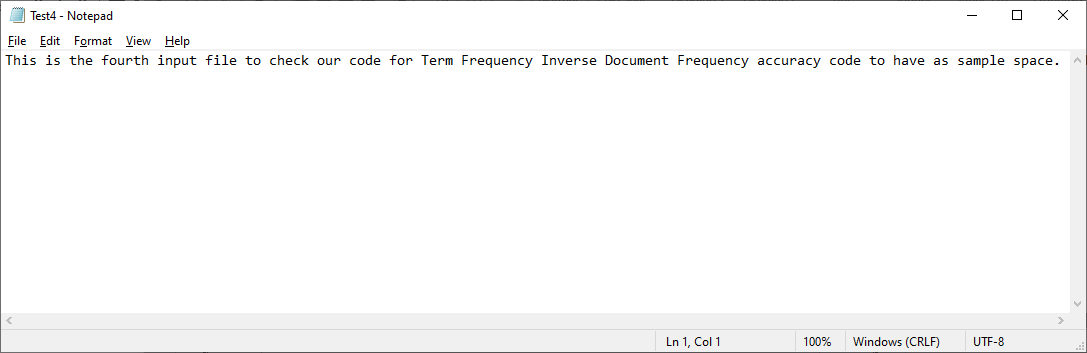
**Text File 2:**

****

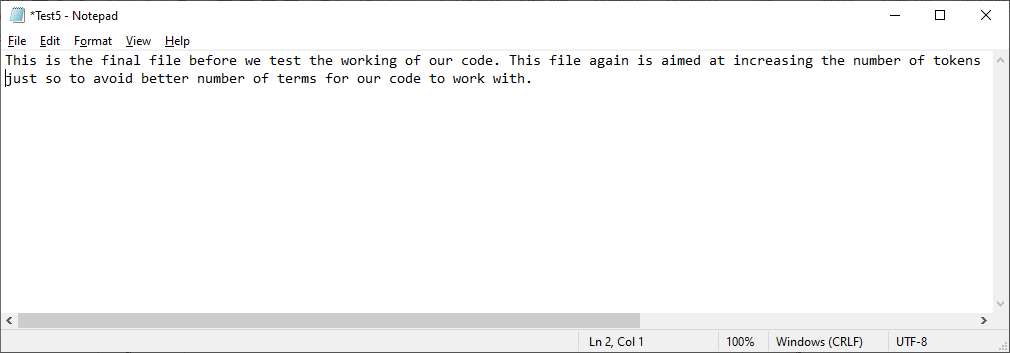
**Text File 3:**



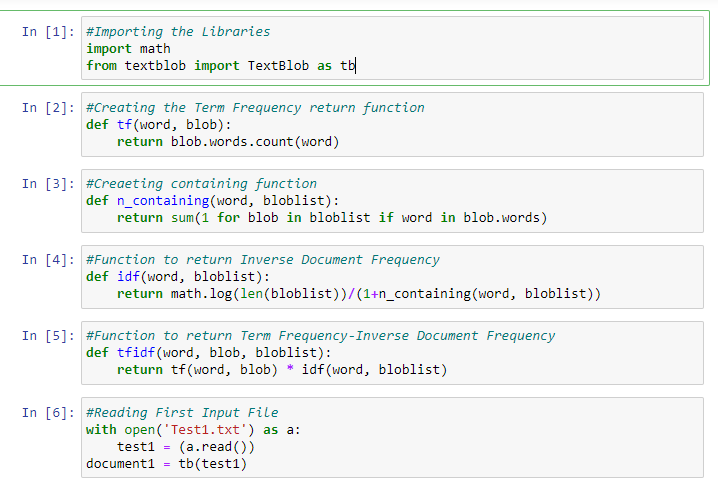
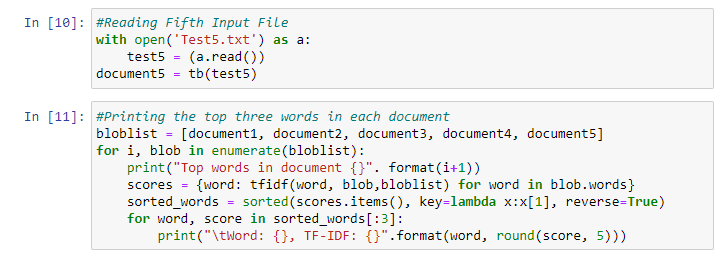
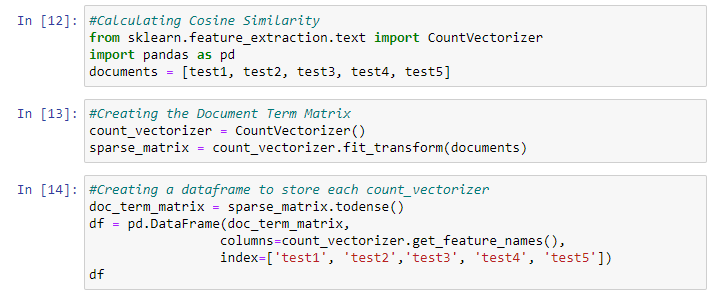
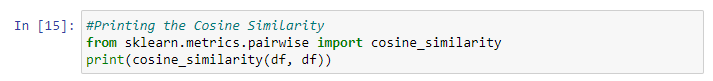
**Text File 4:**



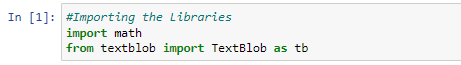
**Text File 5:**



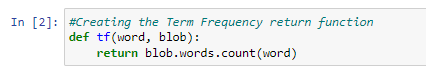
**Code:**

 ****   

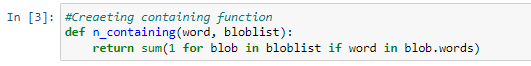
**Code Snippets and Outputs:**



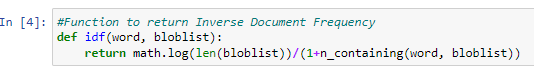
Here we are importi8ng the necessary Libraries



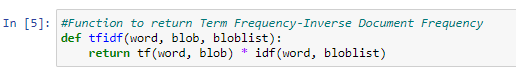
Here we are creating the Term Frequency return Function which takes word and blob as attributes.



Here we are now creating the n\_containing Function wwhich takes words and bloblist as attributes.



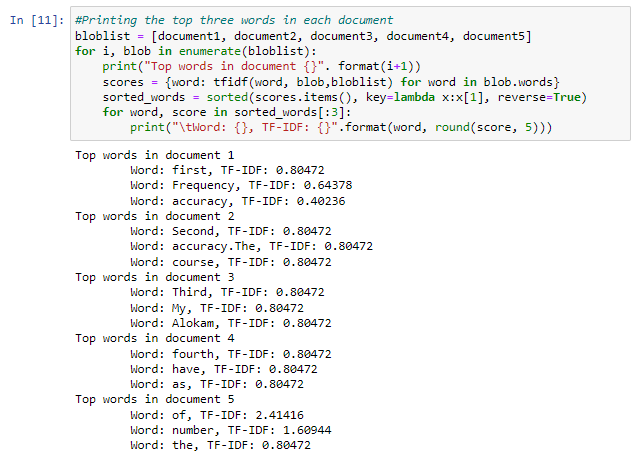
A function named idf is created inorder to Inverse the Document Frequency



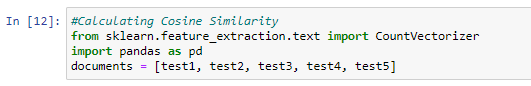
Here we create a Function named ifidtf to return Term Frequency-Inverse Document Frequency



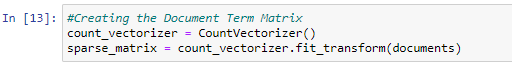
Here we are reading all the 5 Input Text Files(i.e, Test1.txt, Test2.txt, Test3.txt, Test4.txt, Test5.txt)

****

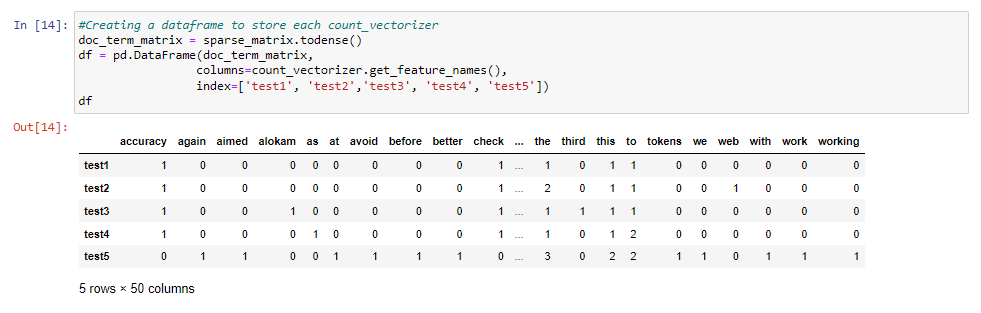
Here we’ve printed the top words in every document. We’ve printed only top 3 words and the TF-IDF values of them in the same line with the word/term.



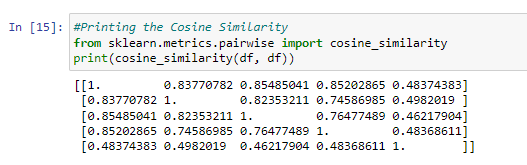
Here we are Calculating the Cosine Similarity of all the Input Text files.



Here we’ve created count vector which contains the frequency of each word of each document. This is for finding the Cosine Similarity.



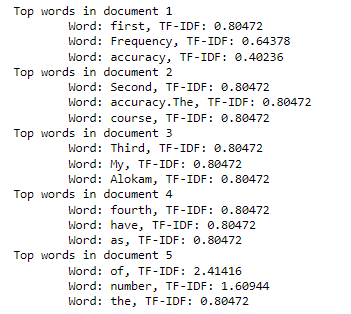
Here we’ve combined the count vectors of every document into Pandas Data Frame.



Here we printed Csonne Similarity of Every Document

**Results and Output**

Top words in Each Input Text file:

****

Cosine similarity

