Assignment
Map-Reduce

Design Lab
31st January 2024

Important Instructions

1. **Submission Rule:** Python code for above functionalities must be compressed as zip and named "**<RollNo>.zip"**. For each query, make a directory named "**Query<no.>"**. Your files must be inside the respective directories. Strictly adhere to this naming convention. Submissions not following the above guidelines will attract penalties.
2. **Code error:** If your code doesn't run or gives error while running, you will be awarded with zero mark. Your code must run correctly on **a linux machine**.
3. **Plagiarism Rule:** If your code matches (more than 50%) with another student's code, all those students whose codes match will be awarded with zero marks (may be with -ve marks too depending on the situation) without any evaluation. Therefore, it is your responsibility to ensure that neither you copy anyone's code nor anyone is able to copy yours.
4. The mapper routine can pass over the network.txt file only once. You can not use any intermediate file structure or data structure to save data. Python dictionaries should not be used in the mapper routine. However, you can use one dimensional arrays up to 10 in the mapper or reducer, and dictionaries of size up to 10 in the reducer only. Note that these must be limited to one dimension only. The reducer can't access the data file. So, the mapper should just go through the data and generate a series of key-value pairs which will then be passed to the suitable sort command. Finally the reducer should just go through the stream of tokens once to give the required output. Any violation will attract a penalty of upto 100% of the marks assigned.

This assignment is on map-reduce, which is a distributed and scalable way of extracting/mining required information from multiple datasets stored on multiple servers. Follow the tutorial to understand how you can design mapper and reducer for specific queries/operations.

Dataset

We will be using a dataset on a Facebook (FB) network with **3,892** users. Please download the data "network.txt". You have to extract required information from this data. Below is a sample of data available in the file.

(**network.txt**)

```
0,1838
0,1744
0,14
0,2543
1,1009
1,1171
1,1465
```

where

- Each line represents an **undirected edge** between two users indicated by two node-ids separated by a comma. So, no pair of nodes are repeated in the file.

Sample code to read the files

```
fp=open("network.txt")
for line in fp:
    l_arr=line.split(",")
    node1=l_arr[0]
    node2=l_arr[1]
```

Queries

The queries are to be implemented in the mapper and reducer phases. Some of them may give empty results. You need to implement the following queries in this assignment. The queries have an imaginary backstory to help you find a real world perspective in this assignment.

1. You are the CEO of a car company. You want to run an advertisement campaign on FB for your newly launched car. An FB official tells you that if you choose to send the ads to all users each with at least 20 friends, then they will charge you INR 100 per user. You want to calculate how much you will have to spend for your ad campaign if you choose to float ads only for users with at least 20 friends. Write mapper and reducer routines for this.
   (**Hint**: *Count all the users (node-ids) with at least 20 friends using mapper and reducer, and then multiply 100 with that before printing the output*) **[30]**
2. The FB official also gives you a free welcome offer valid only for once. This offer is for you to judge the impact of advertising on FB. They say they will send your ad to any 10 users as per your selection. You are a very intelligent business person. So, you think that if you send your ads to users who have a large number of friends, then your ad will easily spread through FB shares, and you won't have to pay anything for the paid ad campaign. So, you want to find the top-10 users (node-ids) with the highest number of friends (order doesn't matter). Write mapper and reducer routines for this. **[30]**
3. After you exploit the free offer, immediately FB realizes the flaw in the offer, and discontinues it. However, your ad has already gone to the top-10 users. Now, FB being a very manipulative company, decides to hide your ad from the newsfeed of the users who are connected to the top-10 users, so that even if the top-10 users share your ad, it won't be visible to others and it won't spread. So, FB wants to find out the users who have friendship with at least one out of the top-10 users. Write mapper and reducer routines for this.
   (**Hint**: *the mapper can use the list obtained in the last query*) **[40]**

How to run and test your code:

```
cat network.txt | python mapper.py | sort | python reducer.py  > result.txt
```

Or just `python mapper.py | sort | python reducer.py > result.txt`

Explanation on the above commands:

1. "cat" is a linux command to print the contents of a file on the console.
2. The pipe operator (|) directs the output of the previous command to the next command.
3. "sort" is a linux command to sort the input lexicographically.
4. ">" can be used to save the standard output in a file.

Deliverables:

1. 3 sets of python codes (mapper.py and reducer.py) one each for query,
2. 3 result.txt each for one query.
3. 1 README.txt explaining your approach to solve each of the query.

==========