CS6350
Big data Management Analytics and Management
Fall 2019
Homework 2
Submission Deadline: 10th Oct, 2019

In this homework, you will use spark (spark, spark dataframe, spark sql) to solve the following problems.

Q1

Write a spark script to find total number of common friends for any possible friend pairs. The key idea is that if two people are friend then they have a lot of mutual/common friends.

For example,
Alice's friends are Bob, Sam, Sara, Nancy Bob's friends are Alice, Sam, Clara, Nancy Sara's friends are Alice, Sam, Clara, Nancy

As Alice and Bob are friend and so, their mutual friend list is [Sam, Nancy]
As Sara and Bob are not friend and so, their mutual friend list is empty. (In this case you may exclude them from your output).

Input files:
1. soc-LiveJournal1Adj.txt

The input contains the adjacency list and has multiple lines in the following format:
<User><TAB><Friends>

Here, <User> is a unique integer ID corresponding to a unique user and <Friends> is a comma-separated list of unique IDs (<User> ID) corresponding to the friends of the user. Note that the friendships are mutual (i.e., edges are undirected): if A is friend with B then B is also friend with A. The data provided is consistent with that rule as there is an explicit entry for each side of each edge. So when you make the pair, always consider (A, B) or (B, A) for user A and B but not both.

Output: The output should be in the following format:
<User_A>, <User_B><TAB><Mutual/Common Friend Number>
where <User_A> & <User_B> are unique IDs corresponding to users A and B (A and B are friend). < Mutual/Common Friend Number > is total number of common friends between user A and user B.

Q2.

Please answer this question by using data sets below.
1. soc-LiveJournal1Adj.txt

2. userdata.txt
The userdata.txt consists of column1 : userid
column2 : firstname column3 : lastname column4 : address column5: city column6 :state

column7 : zipcode column8 :country column9 :username
column10 : date of birth.

Find top-10 friend pairs by their total number of common friends. For each top-10 friend pair print detail information in decreasing order of total number of common friends. More specifically the output format can be:

<Total number of Common Friends><TAB><First Name of User A><TAB><City of User A> <TAB><Age of User A><TAB><First Name of User B><TAB><City of User B><TAB><Age of User B>
…

Q3.
In this question, you will learn how to solve problems using Apache Spark. Please use Apache Spark to derive some statistics from **Yelp Dataset**.

Data set info:

The dataset files are as follows and columns are separate using '::'
business.csv.
review.csv.
user.csv.

Data set Description.
The data set comprises of three csv files, namely user.csv, business.csv and review.csv.

business.csv file contain basic information about local businesses, and it contains the following columns: "business_id"::"full_address"::"categories"

'business_id': (a unique identifier for the business)
'full_address': (localized address),
'categories': [(localized category names)]

review.csv file contains the star rating given by a user to a business. Use user_id to associate this review with others by the same user. Use business_id to associate this review with others of the same business. This file contains the following columns:

"review_id"::"user_id"::"business_id"::"stars"
'review_id': (a unique identifier for the review)
'user_id': (the identifier of the reviewed business),
'business_id': (the identifier of the authoring user),
'stars': (star rating, integer 1-5), the rating given by the user to a business.

user.csv file contains aggregate information about a single user. This file contains the following columns
"user_id"::"name"::"url"
user_id': (unique user identifier),

'name': (first name, last initial, like 'Matt J.'), this column has been made anonymous to preserve privacy
'url': url of the user on yelp

Note: :: is column separator for all the files.

List the 'user id', 'name' and 'rating' of users that reviewed businesses classified as “Colleges & Universities” in list of categories.
Required files are 'business' and 'review'.

Sample output

User id	Name	Rating
Tpmvufw1eea1DrjLAY2jLg	Theodore J.	4.0

Q4
List the business_id , full address and categories of the Tail 20 (worst rated) businesses located in “NY” using the average ratings.

This will require you to use review.csv and business.csv files.

Sample output:

business id	full address	categories	avg rating
A_Fm4v2...	16 Division StCohoes…	List[‘Food’, ‘Coffee & Tea’]	1.25

Submission ::
You have to upload your submission via e-learning before due date.
Please upload the following to eLearning:
1. source files
2. output of your program