Assignment 5

Due Sunday, March 20th
Professor Yanlei Diao

This is a programming assignment based on MapReduce, HBase, and Hive on the Telecom Cluster. Each student should have a user name and password on the cluster already.

Schema and Queries

We will use the DBLP dataset from homework 3.

Exercise 1. Programming using MapReduce

We will use table *paperauths*(*paperid, authid*) in that dataset and consider the following two queries in this assignment.

- **Q1**: Find the top *K* authors who rank the highest according to the number of unique coauthors they have. Here, *K* is an input argument to the query.
- **Q2**: For each author, find the top *K* coauthors with whom this author has published the most, and list the number of papers between this author and her coauthors. Again, *K* is an input argument to the query.

The task is to implement Q1 and Q2 using MapReduce, both of which may involve multiple rounds of map and reduce. Each job (a round of map and reduce) should take at least three parameters: **input directory**, **output directory**, **number of reducers**. Since your code should work for arbitrary value of *K*, at least one job in each query should have ***K*** as a parameter. Feel free to add more parameters if needed.

The input data file has already been loaded to a common directory.
`/infres/ir430/bd/dsm2/datasets/dblp_text/`
For instance, the tab-delimited file for `paperauths.tsv` is located at:
`/infres/ir430/bd/dsm2/datasets/dblp_text/paperauths.tsv`

You will just have to import it into your Hadoop File System (HDFS).

Output Format

Since your result will be validated against the true result, make sure that you output the result in the same format, as specified below:

- Q1: *authorid* and *number of unique coauthors*, one line per author, output in the decreasing order of the coauthor count. Please separate authorid and number of unique coauthors by tab. Example output for *K*=2:
  31 30

286 28
400 28

- Q2: *authorid* and <*coauthorid, number of papers they have published*> pairs, one line per author, output in the increasing order of authorid. In each line, output the pairs in the decreasing order of the paper count. Please separate authorid and pairs by tab, and separate elements in each pair by a comma and a space. Example output for *K*=2:
  0 <33, 20> <32, 18>
  1 <72, 4> <49, 2> <57, 2>
  …

In the event of a tie, for Q1, list all the authors that have the same coauthor count in the increasing order of authorid (e.g., the second and the third lines in the example output of Q1); for Q2, list all the coauthors that have the same paper count in the increasing order of coauthorid (e.g., the second line in the example output of Q2). Make sure to include all the coauthors involved in the tie, even if they will make the number of answers exceed *K*.

MapReduce Jobs for a Query Plan

For each particular query, we combine the relational operators in appropriate ways to construct the **map** and **reduce** functions as we discussed in class. Sometimes, we may need multiple rounds of map and reduce in a query program. The output of one round is written to the file system and will later be read as input to the next round.

Submission

Please submit your **source code** on the Telecom Cluster under the path "$ur_home_dir/A5/" before the due date.

Please also hand in a **report** through Moodle, including:

1. the query plan (operator tree) you considered for each query,
2. brief description of each map/reduce function in your code,
3. the configuration instruction in order to run your code, and
4. (optional) the output file for the specific *K* value you've picked.

It's possible to use many languages with Hadoop, BUT you are expected to code in Java.