Lab 6

In this lab, we will use Hadoop Streaming to write a MapReduce program for computing the cross product of two tables in a database.
Recall the following example from the class slides. We have the following tables:

Professors:

t_num	t_name
101	Smith
105	Jones
110	Fong

Courses:

c_num	c_name
514	Intro_to_DB
513	Intro_to_OS

The cross product of these two tables is:

t_num	t_name	c_num	c_name
101	Smith	514	Intro_to_DB
105	Jones	514	Intro_to_DB
110	Fong	514	Intro_to_DB
101	Smith	513	Intro_to_OS
105	Jones	513	Intro_to_OS
110	Fong	513	Intro_to_OS

The tables will be given to you as 2 input files, one for each table, in the format (tablenumber,id,name). For the example above, the input would be:

professors.txt:

```
1,101,Smith
1,105,Jones
1,110,Fong
```

courses.txt:

```
2,514,Intro_to_DB
2,513,Intro_to_OS
```

The reducer output should be the cross product of the two tables in the form (t_num,t_name,c_num,c_name). For the example above, the output should be:

101, Smith, 514, Intro_to_DB
105, Jones, 514, Intro_to_DB
110, Fong, 514, Intro_to_DB
101, Smith, 513, Intro_to_OS
105, Jones, 513, Intro_to_OS
110, Fong, 513, Intro_to_OS

Assignment:

You must get into groups of 2-4 students.

You will write map and reduce functions in python that compute the cross product between the professors and courses tables. We have provided you with professors.txt and courses.txt input files that contain professor/course data for the Math department at NYU.

1. As a group, decide on a format for the intermediate files that the map.py function will output and the reduce.py function will take as input. Sketch out how the map and reduce functions will work.

   Note: In the interest of simplicity, you may assume that you can hard-code in the number of rows in each table.

2. Within the group, assign 1 or more people to write the map task, and 1 or more people to write the reduce task. Skeleton map.py and reduce.py files, as well as the input files, are under the Lab6 folder on NYU Classes.

   Suggestion: You should first test your code locally before running it using Hadoop, since debugging will be much easier. This will require changing the map.py and reduce.py code to read/write from a text file rather than sys.stdin.

3. When you are finished, you can test your code with Hadoop streaming using HPC's dumbo machine (see lab 1). (Note: if too many people logging in to dumbo results in access problems, you can also run this locally on your laptop, or on an EMR cluster).

   Note: You should use 1 reducer. On dumbo, this means running your file using the command

   ```
   hjs -D mapreduce.job.reduces=1 -files map.py,reduce.py -mapper
   map.py -reducer reduce.py -input /user/your_netid/professors.txt
   -input /user/your_netid/courses.txt -output
   /user/your_netid/table.output
   ```

   On an EMR cluster, put `-D mapreduce.job.reduces=1` in the arguments textbox when you add a step.

Submission: Submit the file you worked on, either map.py or reduce.py, to NYU classes. In the submission text box, put the netIDs of the other members of your group. Due Monday, March 21, at 12pm.