Virginia Tech. **CS 4604 – Introduction to DBMS**
Computer Science **Spring 2015, Prakash**

Homework 5: Map-Reduce

(due March 30th 4:00pm, in class—hard-copy please)

Reminders:

a. Out of 100 points. Contains 3 pages.
b. Rough time-estimates: 6-8 hours.
c. Please type your answers. Illegible handwriting may get no points, at the discretion of the grader. Only drawings may be hand-drawn, as long as they are neat and legible.
d. There could be more than one correct answer. We shall accept them all.
e. Whenever you are making an assumption, please state it clearly.
f. Each HW has to be done *individually*, without taking any help from non-class resources (e.g. websites etc).

Q1. Map-Reduce [100 points]

In this question, we will use Map Reduce to figure out the 2-grams in a large text corpus given the all the distinct 4-grams from the text corpus. The idea is to convince you that using Hadoop on AWS has now really become a low-enough cost/effort proposition (compared to setting up your own cluster). You can use one of Java/Python/Ruby to implement this question. You are free to use Hadoop Streaming as well if you want.

Familiarize yourself with AWS (Amazon Web Services). **Read the set-up guidelines posted on the website to set up your AWS account and redeem your free credit ($100)---do this early!**
Link: http://courses.cs.vt.edu/~cs4604/Spring15/homeworks/hw5/AWS-setup.pdf

The pricing for various services provided by AWS can be found at http://aws.amazon.com/pricing/. The services we would be primarily using for this assignment are the Amazon S3 storage, the Amazon Elastic Cloud Computing (EC2) virtual servers in the cloud and the Amazon Elastic MapReduce (EMR) managed Hadoop framework. Play around with AWS and try to create MapReduce job flows (not required, or graded) or try the sample job flows on AWS.

The questions in this assignment will ideally use up only a **very small fraction of your $100 credit (around $10 or $11)**. AWS allows you to use up to 20 instances total (that means 1 master instance and up to 19 core instances) without filling out a "limit request form". For this assignment, **you should not exceed this quota of 20 instances**. You can learn about these instance types by going through the extensive AWS documentations. Of course, after you are done with the HW, feel free to use your remaining credits for any other fun computations/applications you may have in mind! These credits are applicable more generally for AWS as a whole, not just MapReduce.

We will use data from the Google Books n-gram viewer corpus. N-grams are fixed size tuples of items. In this case the items are words extracted from the Google Books corpus. The n specifies the number of elements in the tuple, so a 5-gram contains five words. This data set is freely available on Amazon S3 in a Hadoop friendly file format and is licensed under a Creative Commons Attribution 3.0 Unported License. The original dataset is available from http://books.google.com/ngrams/.

The subset we will be using for this assignment is a subset of the 4-gram English 1M dataset in the following S3 bucket (directory) and is accessible to all:

s3n://cs4604-2015-vt-cs-data/eng-1m/

The data is in a simple txt file, and each row of the dataset is formatted like:

```
ngram TAB year TAB match_count TAB page_count TAB volume_count NEWLINE
```

(note that the file is TAB delimited). For example, 2 sample lines in our dataset could be:

```
analysis is often described  1991  10   1   1
analysis is often described  1992  30   2   1
```

where 'analysis is often described' is a 4-gram and line tells us that it occurred 10 times in the year 1991 in 1 book in the Google books sample, 30 times in the year 1992 and so on.

Refer to our setup guidelines to see how to set this data as input to your MapReduce job (Section 6 in the guidelines). We have provided a screenshot to configure the EMR cluster, which demonstrates how to access input data from some given bucket (here our bucket is the one given above).

Also, here is a step-by-step process of how to run an example 'WordCount' job on AWS: http://courses.cs.vt.edu/~cs4604/Spring15/homeworks/hw5/SBS-AWS-Wordcount.pdf

Q1.1. (50 points) Plot the frequency distribution for the occurrence counts of the 4-grams i.e. a plot where the x-axis is the occurrence count (say k), and y-axis is the number of 4-grams which occur k times. Occurrence count is the just the *total* number of times a particular n-gram has occurred over all the years in the sample.
Hint: It will be easiest if you write a MR job to pull out just the occurrence information from the dataset; download it to your local machine and then compute and plot the distribution again locally on your machine.

Q1.2. (40 points) Write a MapReduce job to output all 2-grams using the same dataset. Store the output (i.e. the 2-gram dataset) in a bucket in S3.

Q1.3. (10 points) Run the same code you used for Q1.1 for the dataset generated above, and similarly plot the frequency distribution for the occurrence counts of the *2-grams* this time. i.e. a plot where the x-axis is the occurrence count (say k), and y-axis is the number of 2-grams which occur k times.

Code Deliverables: For Q1.1: Give the mapper and reducer files in addition to the plot. For Q1.2: Give the mapper and reducer for computing the 2-grams from the dataset. For Q1.3: Just include your plot. Zip all of these as YOUR-LASTNAME.zip and send it to Elaheh and Yao with the subject 'CS 4604 HW5-Code'. **Also copy-paste these in your hard copy.**

***Note*:** In order to avoid extra charges, after you are done with your homework, do not forget to remove all your files in s3 buckets; i.e. the ones generated as output of 4-grams and 2-grams counts and any files you have uploaded.