**HW1**

PART 1. Run every Hadoop File System Shell commands, and take the screenshot of every command you execute and place into a Word Document. Learn every shell command rather than blindly copy-pasting. Briefly write what each command does into the same document after or before the screenshot.

<http://hadoop.apache.org/docs/r2.7.3/hadoop-project-dist/hadoop-common/FileSystemShell.html>

PART 2. Download and Copy all the files to a folder in HDFS

<http://msis.neu.edu/nyse>

   Use the NYSE files in HDFS to find the average price of stock\_price\_high values for each stock using MapReduce in a Java application.

PART 3. Download the following dataset and Copy all the files to a folder in HDFS

MovieLens 10M - Stable benchmark dataset. 10 million ratings and 100,000 tag applications applied to 10,000 movies by 72,000 users.

<http://grouplens.org/datasets/movielens>

Write a MapReduce to find the number of males and females in the movielens dataset

PART 4. Write a MapReduce to find the number of movies rated by different users

PART 5.  Write a MapReduce to find the number of times this website has been accessed by each IP Address using the attached log file.

**HW2**

PART 2. Download and Copy all the files to a folder in HDFS (http://msis.neu.edu/nyse/)

   Write a Java Program to implement PutMerge as discussed in the class to merge the NYSE files in HDFS to find the average price of stock-price-high values for each stock using MapReduce on the single merged-file. Compare the running times of your original program doing MapReduce on multiple files to the modified version that merges all the files to a single file to performMapReduce.

PART 3. Write one short program using each of the classes that extend FileInputFormat<K,V>

(CombineFileInputFormat, FixedLengthInputFormat, KeyValueTextInputFormat, NLineInputFormat, SequenceFileInputFormat, TextInputFormat)

http://hadoop.apache.org/docs/r2.7.3/api/org/apache/hadoop/mapreduce/lib/input/FileInputFormat.html

You could use any input file of your choice. The size of the input files is not important.

PART 4. Create a Writable object that stores some fields from the the NYSE dataset to find

- the date of the max stock\_volume

- the date of the min stock\_volume

- the max stock\_price\_adj\_close

PART 5. Download the following dataset and Copy all the files to a folder in HDFS

MovieLens 10M - Stable benchmark dataset. 10 million ratings and 100,000 tag applications applied to 10,000 movies by 72,000 users.

http://grouplens.org/datasets/movielens/

Write a MapReduce to find the top 25 rated movies in the movieLens dataset

**HW3**

2. Determine the average stock\_price\_adj\_close value by the year.

Choose an implementation in which a Reducer could be used as a Combiner. (discussed in the lecture, and available in the slides).

3. Using the MoviLens dataset, determine the median and standard deviation of ratings per movie.

Iterate through the given set of values and add each value to an in-memory list. The iteration also calculates a running sum and count.

4. Redo Part 3 using Memory-Conscious Median and Standard Deviation implementation as explained in the Slides (MR Summarization Patterns Slides).  
Use a Combiner for optimization.

**HW4**

Implementation of different patterns using mapreduce