

# BIG DATA –Spring 2018

## ASSIGNMENT 2

**Due Date: 16 Feb 2018 till 10:00 PM on google classroom.**

Upload the Source code and the output file on Google Classroom.

### Question 1: COMBINER

In this question you will have to just add combiner to the Assignment 1.

**Hint:** In this particular example the combiner would be same as the reducer, so you have to add following line to run function. Specify your Reducer as the combiner class.

**`job.setCombinerClass(MyReducer.class);`**

### Question 2: COMBINER AND PARTITIONER

You are given a text file consisting of the name of the Movie, its release Year and the movie rating (on a scale of 1-5) given to the movie by IMDB. Your task is to find the maximum rating given by the IMDB each year. The data is for time period 1900-2018.

You have to write code for Mapper, Reducer and Combiner.

In addition to that you must ensure that the number of reducer is 3 and all the movies released in year 1900-1960 should go to Reducer 1, movies with release year 1961-1990 should go to Reducer 2 and movies with release year 1991-2018 should go to the last reducer. **You can accomplish this by using the Custom Partitioner.**

#### Input File

The dark dragon 1929 3.5
Evil Witch 1958 2
Dark Lord 1958 3
SuperMan 1972 4.5
Twister 1986 3.5
Tornado 2010 4
The rise of Aron 1965 2.5
Jumangi 1929 4
Snow Queen 1986 2.5
Red Riding Hood 2012 4.5
Lords 2010 3

#### Output Files

Output File 1	Output File 2	Output File 3
1929 4 1958 3	1965 2.5 1972 4.5 1986 3.5	2010 4 2012 4.5

**Hints: (consult book Hadoop the definitive guide)**

To set the number of reduce task equal to 2 write the following line of code in driver program that is **run** function.

***job.setNumReduceTasks(2);***

You can specify your partitioner let's call it YearPartitioner in run function as follows.

***job.setPartitionerClass(YearPartitioner.class);***

In addition to this you need to specify your Partitioner class and **getPartition** function in it.

```
public static class YearPartitioner extends Partitioner<IntWritable, FloatWritable> {  
    @Override  
    public int getPartition(IntWritable key, FloatWritable value, int numPartitions) {  
        // you code goes here  
        // return the appropriate value  
    }  
}
```