Name:____________________________ **ID:**_____________

Maharishi International University
DEPARTMENT OF COMPUTER SCIENCE
CS 522 Big Data
Final Exam

Question I [30 Points]

This question has three parts. What follows is the same for all three parts:

Input:
You can assume the record has seven "fields". For part (a) and (b) you can assume "getFirst()" and so on. **You cannot assume that for Part(c).** The seven fields are First, Last, Score1, Score2, Score3, Score4 and Department.

Output:
One line of output for each department. It has four fields. The first field is the Department. The remaining three fields are maximum, minimum and the average total points received by some student in that department. Output is in the ascending order of Department.

The following example is to further clarify the question. It is not the entire data. Do not use literals in that example in your code. Do not assume there is only one input-split. Do not assume there is only one reducer.

Sample Input

First	Last	Score1	Score2	Score3	Score4	Department
Jane	Lee	80	60	90	70	CSC
Alex	Cox	40	70	70	80	CSC
Mark	Lui	50	60	90	70	ENG
Adi	Hix	90	50	70	80	CSC
John	Brix	60	70	95	75	ENG
Rick	May	70	85	75	40	ENG

Sample output

```
CSC  300   260   283.33
ENG  300   270   280.0
```

Explanation of the logic (if you were writing a Java program).
Jane has (80+60+90+70) = 300 total points.
Alex has (40+70+70+80) = 260 total points
Adi has (90+50+70+80) = 290 total points
Since these are the CSC students, one of the output line is

```
CSC   300    260    283.33
```

Mark has (50+60+90+70) = 270
John has (60+70+95+75) = 300
Rick has (70 +85+75+40) = 270
Since these are the ENG students, one of the output line is

```
ENG  300    270    280.0
```

Thus the output is

```
CSC   300    260    283.33
ENG  300    270    280.0
```

VERY IMPORTANT: Note that the output is ordered in the ascending order of the Department.

END OF EXAMPLE

Part (a) [10 Points]. Solve the problem with **MapReduce and no in-Mapper Combining**. It is your responsibility to provide Mapper class, Reducer class, Partitioner only if needed and a comparator only if needed.

==

Part (b) [10 Points]. Solve the problem with **MapReduce and in-Mapper Combining**. It is your responsibility to provide Mapper class, Reducer class, Partitioner only if needed and a comparator only if need.

==

Part(c) [10 Points] Solve the problem with a segment of **Scala/Spark core** code. **Do not use SQL or SparkSQL. This is to test your ability to solve using Scala and Spark API and not SQL. If you use SQL, you will receive 0 points.**

Question II [10 Points]

Input record contains Four data items.

GroupID FirstName LastName Score

For this question, you need not parse the record. There are **four helper methods**

getGroupID() returns the GroupID as a String
getFirstName() returns the FirstName as a String
getLastName() returns the LastName as a String
getScore() returns the Score as a double.

You are task is to write a MapReduce program. That is

(a) Mapper class,
(b) Reducer class,

that will output

GroupID FirstName LastName Score GroupAvearge

Corresponding to each input record. (Hint: **The GroupAverage is the same for every student in CSC** and so on. If you consider the sample data below, the GroupAverage for CSC is (80 +90 + 70)/3 = 80.

(c) Comparator for the custom class or classes.
(d) getPartition method that will allow **maximum parallelism without any logical errors.** (Must use hashCode() and % operator.

Your output must be sorted as follows:

GroupID	*Ascending*	*followed by*	*(or Primary sorting)*
FirstName	*Descending*	*followed by*	*(or Secondary sorting)*
LastName	*Ascending*		*(or Tertiary sorting)*

Note: You may write an algorithm with or without in-mapper combining. The choice is yours. Make sure Professor can read your handwriting.

Sample input clarification purpose only. Treat each line as a record.

Input-Split 0:

CSC	Jim	Jones	80
Math	Chris	Cox	65
CSC	Mark	Mayer	90

Input-Split 1:

Math	Bea	Adair	60
Math	Dane	Etna	55
CSC	Mike	Maher	70

--

Sample output for clarification. Only one record shown. There will be one such record for each student.

--

```
        CSC    Mark   Mayer          90        80
```