CSC 3331: Analysis of Algorithms
Project 3: Parallel Programming and MapReduce Framework

Task # 1: Modifying MaxTemperature to become AvgTemperature [30 pt]
Change the MaxTemperatureReducer.java, so that it produces Average Temperature of each year instead of Maximum Temperature. Test your program with the above temp.txt. Submit your updated java files.

Task # 2: Modifying WordCount to become LetteCount [30 pt]
Change the WordCount.java, so that it outputs the number of words that start with the letters 'a' 'b' and 'c'. This means that for every letter we want to count the total number of words that start with these three letters. You need to change both Map and Reduce functions. Once changed, test your code with following input file.

```
ant bear cat dog elephant
iguana bird cow antelope baboon
```

Your Program should output

```
a 2
b 3
c 2
```

Submit your updated java file.

Task # 3: Developing your own MapReduce Application [20 pt]
See the attached input file OrderDB.txt. Each line in this file contains {Order-ID, Customer_id, Order_date, total} where total is the amount of money spent be the customer on that order. Write a Mapreduce program (use wordcount/maxtemperature as skeleton) that will output the total amount spent by each customer considering all the orders. Submit your java file.

Task # 4: [20 pt]

a) Find two pre-built Spark clusters in your Chameleon project. Claster-1 is configured with one compute node with 24 cores, 128 GB memory and cluster-2 is configured with two computer nodes with 24 cores, 128 GB memory.
b) Find the pre-loaded Wikepedia pageview application and two datasets of size 100GB and 200GB.
c) Run the Wikipedia page views application in Chameleon Cloud as demonstrated in the class. Hint: *spark-submit*
d) Try various runtime configurations by using varying parameters such as *--num-executors --executor-memory*, and *--executor-cores.* Measure your runtime performances for every configurations.
e) Also check the performance metrics of a running spark application using Spark's web user interface (UI).
f) Compare and analyze the results that you received while executing steps d) and e) for each runtime configurations in terms of performance (execution time and other Spark related metrics) and cost (1 service unit = 1 core with 1 GB memory).

Write a report detailing your implementation and experimental results and findings along with your supporting arguments. Submit the report along with your code and data.