Big data processing and analytics

February 2, 2023

Student ID ____

First Name ____

Last Name ____

The exam is **open book**

Part I

Answer to the following questions. There is only one right answer for each question.

1. (2 points) Consider the following Spark Streaming application.

```
import …
public class SparkDriver {
    public static void main(String[] args) throws InterruptedException {
            SparkConf conf = new SparkConf().setAppName("Spark Streaming - Question");
            JavaStreamingContext jssc = new JavaStreamingContext(conf, Durations.seconds(10));
            // Define a DStream associated with the TPC socket localhost:9999
            JavaDStream<String> inputDStream = jssc.socketTextStream("localhost", 9999);

            // Part A
            JavaDStream<Integer> resADStream = inputDStream
                            .map(value -> Integer.valueOf(value))
                            .window(Durations.seconds(30), Durations.seconds(10))
                            .reduce((v1,v2) -> Math.max(v1,v2))
                            .filter(value -> value>10);
            // Print the result on standard output
            resADStream.print();

            // Part B
            JavaDStream<Integer> resBDStream = inputDStream
                            .map(value -> Integer.valueOf(value))
                            .reduce((v1,v2) -> Math.max(v1,v2))
                            .filter(value -> value>10)
                            .window(Durations.seconds(30), Durations.seconds(10))
                            .reduce ((v1,v2) -> Math.max(v1,v2))
                            .filter(value -> value>10);
            // Print the result on standard output
            resBDStream.print();

            // Part C
            JavaDStream<Integer> resCDStream = inputDStream
                            .window(Durations.seconds(30), Durations.seconds(10))
                            .map(value -> Integer.valueOf(value))
                            .reduce((v1,v2) -> Math.max(v1,v2))
                            .filter(value -> value>10);
```

```
                // Print the result on standard output
                resCDStream.print();

                // Start the computation
                jssc.start();
                jssc.awaitTerminationOrTimeout(120000);
                jssc.close();
        }
}
```

Which one of the following statements is true?

a) Independently of the content of **inputDStream**, **resADStream** and **resBDStream** contain always the same integer values, while **resCDStream** may contain different integer values with respect to **resADStream** and **resBDStream**.

b) Independently of the content of **inputDStream**, **resADStream** and **resCDStream** contain always the same integer values, while **resBDStream** may contain different integer values with respect to **resADStream** and **resCDStream**.

c) Independently of the content of **inputDStream**, **resBDStream** and **resCDStream** contain always the same integer values, while **resADStream** may contain different integer values with respect to **resBDStream** and **resCDStream**.

d) Independently of the content of **inputDStream**, **resADStream**, **resBDStream**, and **resCDStream** contain always the same integer values.

2. (2 points) Consider the input HDFS folder myFolder that contains the following three files:
   - ProfilesItaly.txt
     - The text file ProfilesItaly.txt contains the following four lines:
       Luca,Rome
       Luca,Rome
       Carmen,Naples
       Luca,Turin
   - ProfilesFrance.txt
     - the text file ProfilesFrance.txt contains the following line:
       Claire,Paris
   - ProfilesSpain.txt
     - the text file ProfilesSpain.txt contains the following two lines:
       Carmen,Barcelona
       Pablo,Barcelona

Suppose that you are using a Hadoop cluster that can potentially run up to 3 instances of the mapper class in parallel. Suppose the HDFS block size is 1024MB. Suppose to execute a MapReduce application for Hadoop that analyzes the content of myFolder. Suppose the map phase emits, overall, the following key-value pairs (the key part is a name while the value part is always 1):

(Luca, 1)
(Luca, 1)
(Carmen, 1)
(Luca,1)
(Claire,1)
(Carmen,1)
(Pablo,1)

Suppose the number of instances of the reducer class is set to 2 and suppose the reduce method of the reducer class sums the values associated with each key and emits one pair (name, sum values) for each key if the sum is greater than 2. Specifically, suppose the following pair is emitted, overall, by the reduce phase:

(Luca, 3)

Considering all the instances of the reducer class, overall, how many times is the **reduce method** invoked?

a) 1
b) 2
c) 3
d) 4

Part II

HousePowerConsumption (HPC) is an international company that monitors the power consumption of private houses around the world. HPC computes a set of statistics about the monitored houses. The analyses are performed by considering the following input data sets/files.

- Houses.txt
    - Houses.txt is a text file containing the list of houses monitored by HPC. Each line of Houses.txt is associated with one house and contains its profile. The number of monitored houses is more than 1000000.
    - Each line of Houses.txt has the following format
        - HouseID,City,Country,SizeSQM

          where *HouseID* is the house identifier, *City* and *Country* are the city and country in which the house is located, respectively, and SizeSQM is the size of the house in square meters.
        - For example, the following line

          House102,Turin,Italy,120

          means that the house identified by ***House102*** is located in the city of ***Turin*** (***Italy***) and its size is ***120*** square meters.

- DailyPowerConsumption.txt
    - DailyPowerConsumption.txt contains information about the daily power consumption of the houses under analysis in the last 20 years. Suppose there is one line for each pair (house, date) considering all houses monitored by HPC and all the dates of the last 20 years.
    - Each line of DailyPowerConsumption.txt has the following format
        - HouseID,Date,kWh
          where *kWh* is the kilowatt-hours consumed by the house with id *HouseID* on the date *Date*.
          Each line of DailyPowerConsumption.txt is uniquely identified by the primary key (HouseID,Date). Hence, each combination (HouseID,Date) occurs at most one time in DailyPowerConsumption.txt.
          The format of the date is YYYY/MM/DD.
        - For example, the following line

          House102,2022/12/21,12

          means that the power consumption of the house with HouseID ***House102*** was ***12*** kWh on ***December 21, 2022***.

Exercise 1 – MapReduce and Hadoop (8 points)

Exercise 1.1

The managers of HPC are interested in performing some statistics.

Design a single application, based on MapReduce and Hadoop, and write the corresponding Java code, to address the following point:

1. *City with the highest number of small houses.* Selects the city with the highest number of small houses (a house is classified as a “small house” if its size is less than 60 square meters). If there is more than one city associated with the highest number of small houses, the first one in alphabetical order is selected. Store the name of the selected city in the output HDFS folder.

Suppose that the input is Houses.txt and has been already set. Suppose that also the name of the output folder has been already set.

- Write only the content of the Mapper and Reducer classes (map and reduce methods. setup and cleanup if needed). The content of the Driver must not be reported.
- Use the following two specific multiple-choice questions (**Exercises 1.2 and 1.3)** to specify the number of instances of the reducer class for each job.
- If your application is based on two jobs, specify which methods are associated with the first job and which are associated with the second job.
- If you need personalized classes, report for each of them:
    - the name of the class
    - attributes/fields of the class (data type and name)
    - personalized methods (if any), e.g., the content of the toString() method if you override it
    - do not report the get and set methods. Suppose they are "automatically defined"

Exercise 1.2 - Number of instances of the reducer - Job 1

Select the number of instances of the reducer class of the first Job

□ (a) 0

□ (b) exactly 1

□ (c) any number >=1 (i.e., the reduce phase can be parallelized)

Exercise 1.3 - Number of instances of the reducer - Job 2

Select the number of instances of the reducer class of the second Job

□ (a) One single job is needed

□ (b) 0

□ (c) exactly 1

□ (d) any number >=1 (i.e., the reduce phase can be parallelized)

Exercise 2 – Spark (19 points)

The managers of HPC asked you to develop one single application to address all the analyses they are interested in. The application has four arguments: the input files Houses.txt and DailyPowerConsumption.txt, and two output folders "outPart1/" and "outPart2/", which are associated with the outputs of the following points 1 and 2, respectively.

Specifically, design a single application, based on Spark, and write the corresponding code, to address the following two points:

1. *Countries without houses that are characterized by a high average daily consumption in the year 2022*. The first part of this application considers only the year 2022 and selects the countries that are never associated with houses with a high average daily consumption in the year 2022. A house is considered a house with a high average daily consumption in the year 2022 if the average daily consumption of that house in the year 2022 is greater than 30 kWh. Store the selected countries in the first HDFS output folder (one country per line).

   Note. Suppose that in DailyPowerConsumption.txt there is one line for each pair (house, date) considering all houses monitored by HPC and all the dates of the last 20 years.

2. *The number of cities with many houses with high power consumption in the year 2021 for each country*. This second part of the application considers only the consumption in the year 2021 and computes for each country the number of cities each one with at least 500 houses with high annual consumption in the year 2021. Specifically, a house is classified as a "house with a high annual consumption in the year 2021" if its annual consumption in the year 2021 is greater than 10000 kWh. Store the result in the second output folder (one country per output line). The output format is "country,number of cities each one with at least 500 houses with a high annual consumption in the year 2021 for that country". Save also the information for the countries with no cities with at least 500 houses with a high annual consumption in the year 2021. In those cases, the output line will be "country,0".

- You do not need to report imports. Focus on the content of the main method.
- Suppose both JavaSparkContext ***sc*** and SparkSession ***ss*** have been already set.
- If you need personalized classes, report for each of them:
  - the name of the class
  - attributes/fields of the class (data type and name)
  - personalized methods (if any), e.g., the content of the toString() method if you override it
  - do not report the get and set methods. Suppose they are "automatically defined"