Lab 3 - MapReduce Average Temperature Lab

----

o Submit your *own work* on time. No credit will be given if the lab is submitted after the due date.
o Note that the completed lab should be submitted in .zip format only.

Description of the given Input Dataset (NCDC-Weather.txt)

➤ For these problems, the data we will use is from the National Climatic Data Center (NCDC). The data is stored using a line-oriented ASCII format, in which each line is a record. The format supports a rich set of meteorological elements, many of which are optional or with variable data lengths. For simplicity, we focus on the basic elements, such as temperature, which are always present and are of fixed width.

➤ Example below shows sample lines with some of the salient fields annotated. The line has been split into multiple lines to show each field; in the real file, fields are packed into one line with no delimiters.

> 0067011990999991950051507004+68750+023550FM-12+038299999V0203301N00671220001CN9999999N9+00001+99999999999
>
> 0043011990999991950051512004+68750+023550FM-12+038299999V0203201N00671220001CN9999999N9+00221+99999999999
>
> 0043011990999991950051518004+68750+023550FM-12+038299999V0203201N00261220001CN9999999N9-00111+99999999999
>
> 0043012650999991949032412004+62300+010750FM-12+048599999V0202701N00461220001CN0500001N9+01111+99999999999
>
> 0043012650999991949032418004+62300+010750FM-12+048599999V0202701N00461220001CN0500001N9+00781+99999999999

➤ Format of this National Climatic Data Center (NCDC) record is as follows:

- 0057
- 332130   # USAF weather station identifier
- 99999   # WBAN weather station identifier
- **19500101**   **# observation date** (substring(15, 19))

- 0300   # observation time
- 4
- +51317   # latitude (degrees x 1000)
- +028783   # longitude (degrees x 1000)
- FM-12
- +0171   # elevation (meters)
- 99999
- V020
- 320   # wind direction (degrees)
- 1   # quality code
- N 0072
- 1 00450   #sky ceiling height (meters)
- 1   # quality code
- C
- N 010000   #visibility distance (meters)
- 1   # quality code
- N
- 9
- **-0128**   **# air temperature (degrees Celsius x 10)** (substring(87, 92))
- 1   # quality code (any one of [01459] is good quality)
- -0139   # dew point temperature (degrees Celsius x 10)
- 1   # quality code
- 10268   # atmospheric pressure (hectopascals x 10)
- 1   # quality code

Lab Questions

---

1. **Write a basic MapReduce java program without combiner or in-mapper combining to calculate the average temperature per year.** (Remember to divide the temperature by 10)

2. **Write a MapReduce java program with *combiner (no in-mapper combining)* to calculate the average temperature per year.**
   Hint: As this is "Average" problem, default combiner cannot be used. So you'll need to write a Custom Combiner. Take help from the pseudo code given in the lecture slides.
   You'll need to create a custom [Writable] object which will be passed around from the mapper to combiner to reducer.

3. **Write a MapReduce java program with *in-mapper combining* design pattern to calculate the average temperature per year.**

   Hint: Take help from the pseudo code that you've written in the Assignment 4 – Q1 problem.

4. **Modify the above program by writing your own sorting routine so that the output file will show the latest year first**. (Years should be in descending order)

   Hint: Create a Custom Class by extending WritableComparable and override compareTo method. This custom object will be passed around from mapper as Key.

5. **Now, we need to use 2 reducers. So, create a Custom Partitioner class which will send all the years less than 1930 to Reducer 1 and rest of the years to Reducer 2.**
   (Remember, partitioner will not work in local mode of your VM!)

   Hint: Create a custom partitioner class by extending HashPartitioner and override *getPartition* method. Then use the *setPartitionerClass* method of Job to set the Partitioner for your job in the driver code.

Submit the Java files and output files for all the problems.
Also submit the commands which you used to run the jar file of the programs in pseudo-distributed mode.