Homework 3

ANALYZING AIR QUALITY DATA COLLECTED ACROSS THE UNITED STATES USING MAPREDUCE
VERSION 1.1

DUE DATE: Wednesday October 29th, 2025 @ 8:00 pm

OBJECTIVE

The objective of this assignment is to gain experience in developing MapReduce programs. As part of this assignment, you will be working with data collected from the EPA's Air Quality System (AQS). You will be developing MapReduce programs that parse and process recordings of temperature and criteria gas levels at various outdoor monitors.

You will be using Apache Hadoop (version 3.4.0) to implement this assignment. Instructions for accessing datasets will be released in Canvas alongside this assignment.

You are required to work alone on this assignment. Use of GenAI tools is expressly prohibited; see the textbox below. This assignment accounts for **10% of your course grade** and may be modified to clarify any questions (and the version number incremented), but the crux of the assignment and the distribution of points will not change.

Generative AI Use and Consequences

Use of AI tools such as ChatGPT, Claude, Github Co-Pilot, or anything of their kind to write or "improve" your code or written work at *any* stage is prohibited; this includes the ideation phase. It is your responsibility to ensure that you don't have the GitHub Co-Pilot extension installed in your IDE; assignment solutions generated by Co-Pilot aren't written by you. Turning in code or an essay written by generative AI tools will be treated as turning in work created by someone else, namely an act of plagiarism and/or cheating. At a minimum, this will result in a 100% deduction (i.e., you will receive a –10/10). To ensure fairness and maintain integrity, grading will also include code reviews, interviews, and on-the-spot code modifications.

Ultimately, you will get out of the class what you put in. Simply copying and pasting code from generative AI tools is not only unethical, it robs you of the chance to learn. Here are four reasons why these generative AI tools undercuts your own education:

1. They take away the struggle that leads to understanding. They rob you of the ability to think and learn the concepts for yourself. Solving problems yourself is how concepts stick. If the AI does the work, what's left for you to learn?

2. You will struggle with the in-classroom quizzes and exams where you will not have access to these tools.

3. Yes, AI tools will become an important part of a software engineer's workflow. But to use them effectively later, you first need solid expertise in the subject matter; and, that only comes from practicing *without* them.

4. These tools are prone to generating imperfect or even incorrect solutions, so trusting them blindly can lead to bad consequences.

1 Cluster setup

For this assignment, the Hadoop cluster with HDFS running on every node has been already set up for the auto-grader. Datasets have already been staged too. Your programs will process the staged datasets; data locality will be preserved by the MapReduce runtime. We have included a `build.gradle` file at the bottom of this assignment that you should use.

2 Air Quality Dataset

The dataset contains daily measurements of air quality (AQI) readings from various monitors around the United States. The assignment will use AQI data from all 50 states. The datasets can be downloaded from Canvas. Data for each state will be provided in CSV format. Each file will be named "epa_aqi_score_<state>.csv", e.g., "epa_aqi_score_south_dakota.csv". There are two additional files: "states.csv" and "counties.csv". Each AQI CSV has three columns: GISJOIN, aqi_score, epoch_time. You will have to join states.csv and counties.csv with the AQI score datasets to associate a county and a state with each row of the AQI dataset. states.csv has two columns: state, GISJOIN. counties.csv has two columns: county, GISJOIN. You will load the CSVs from `<input_folder>/states/<state_aqi_csv>.csv`, `<input_folder>/states.csv`, and `<input_folder>/counties.csv` where `<input_folder>` is an argument given to your program. In each csv, there is a GISJOIN column. A GISJOIN is a geo-identifier with the following format: GXXXYYYY. The first three numbers (XXX) after the "G" correspond to the state, and the following four numbers (YYYY) correspond to the county. For example, the GISJOIN for Colorado is G008 and the GISJOIN for Larimer County is G0080069. We will provide the list of states you need to join as an argument to your program. You are not allowed to hardcode any values or ids inside your Java code. All values must be read from the HDFS.

1 Analysis of Air Quality Data

You should develop MapReduce programs that process the AQS dataset to answer the following questions. You must write each of your answers to a separate folder in HDFS, e.g., "<output_folder>/q1/", "<output_folder/q2/", etc

Q1.	What were the lowest AQI and highest AQI Days of Week for AQI scores? Your output should be the key pair <best_day> <worst_day>. For example: Tuesday Friday	**1pt**
Q2.	What were the lowest AQI and highest AQI Months for AQI scores? Your output should be the key pair <best_month> <worst_month>. For example: October March	**1pt**
Q3.	What were the lowest AQI and highest AQI Years for AQI scores? Your output should be the key pair <best_year> <worst_year>. For example: 1997 2021	**1pt**
Q4.	What was the **median** AQI score across all selected datasets? Your output should be only the value <median_score>. For example: 43	**1pt**
Q5.	Which 10 counties had the lowest average AQI scores for the year 2020? List each county on a new line **from lowest AQI score to highest**. For example: Gunnison Alamosa Larimer …	**1pt**
Q6.	Which 10 counties had the highest average AQI scores for the year 2020? List each county on a new line **from highest AQI score to lowest**. For example: Mesa Weld Boulder …	**1pt**

Q7	Rank all states by largest increase in average AQI score between the years 2000 and 2020. You compute the average for a state in 2000, then you compute the average in 2020, then you calculate the difference, then you sort the states from largest increase to smallest. List them on separate lines following the format <State> <Delta>. For example: California 22.05 Washington 17.34 Colorado 13.66 …	**1pt**
Q8	List the biggest one week change for each county in the dataset. List each county alphabetically on a separate line together with its biggest weekly change in average score. For example: Adams 32.51 Alamosa 17.40 Arapahoe 46.34 …	**3pt**

3 Additional Requirements

Grading will be done automatically. You can submit, receive feedback within minutes, make adjustments and resubmit as many times as you want. We will be conducting random interviews after the deadline, and it is important that you are able to explain the method you used to get your answer and why you believe that method accurately answers the question asked.

Try to design your MapReduce jobs as elegantly as possible. This means minimizing the number of jobs and the amount of data transferred between each job. Minimizing the amount of data transferred between the mapper and reducer within each job is also important as it significantly impacts the amount of time the job will take to run.

4 Additional Requirements

There will be a **10-point deduction** if any of the restrictions below are violated.

1. You should not implement this assignment as a stand-alone program.
2. You should not implement this assignment using anything other than Hadoop MapReduce. Implementing your own framework or using a 3rd party framework (that is not Hadoop) to implement this assignment is not allowed.
3. You should not hardcode any values or ids in your Java program.

5 Grading

Homework 3 accounts for 10 points towards your final course grade. The point distribution for this assignment is listed below.

Point Breakdown:

1 point each:	Questions 1-7
3 points:	Question 8

6 Milestones:

You have 4 weeks to complete this assignment. The weekly milestones below correspond to what you should be able to complete at the end of every week.

Milestone 1: You should be able to create a MapReduce Jar file to read data from the HDFS cluster into a MapReduce program and write data from a MapReduce program back to the cluster.

Milestone 2: You should be able to correctly load and join all necessary datasets including those specified in the command line arguments. Your program should be able to answer Q1-Q4.

Milestone 3: Develop MapReduce programs to answer Q5-Q7 and write your answers to the HDFS cluster.

Milestone 4: Complete the MapReduce implementations for Q8.

7 What to Submit

Use **CANVAS** to submit a single .tar file that contains:
- all the Java files related to the assignment (please document your code)
- the `build.gradle` file you use to build your assignment
- a README.txt file containing a manifest of your files and any information you feel the TAs needs to grade your program.

E-mailing the codes to the Professor, GTA, or the class accounts will result in an automatic 1-point deduction.

Filename Convention: All classes should reside in a package called **`cs455.aqi`**. The archive file should be named as FirstName-LastName-HW3.tar. For example, if your name is Foo Bar then the tar file should be named Foo-Bar.tar.

8 How we will test your submission

We will run your code with the following command:
`hadoop jar <your_jar> <state1> <state2> <state3> … <input_folder> <output_folder>`

The first *n* arguments correspond to the list of *n* states that you need to include in your program. All state names will be lowercase, and spaces will be replaced with an underscore, e.g., "South Dakota" becomes "south_dakota".

The second argument is the path to the folder where you should read the datasets from.

The third argument is the path to the folder where you should write your answers to.

9 Version Change History

This section will reflect the change history for the assignment. It will list the version number, the date it was released, and the changes that were made to the preceding version. Changes to the first public release are made to clarify the assignment; the spirit or the crux of the assignment will not change.

Version	Date	Comments
1.0	10/1/2025	First public release of the assignment.
1.1	10/15/2025	We have improved clarity and removed potentially conflicting instructions in sections 1 and 8. Section 10 is a new section that includes a copy of the build-gradle.

10 build.gradle

Please use the below code for your `build.gradle` file.

```
plugins {
    id 'java'
}

group = 'cs455'
version = '2.0-SNAPSHOT'

repositories {
    mavenCentral()
}

dependencies {

    implementation 'org.apache.hadoop:hadoop-client:3.1.2'
}

jar {
    manifest {
        attributes(
            //  main class. Change if your main class is named something different
            'Main-Class': 'cs455.aqi.Questions'
        )
    }
}
```