



Data Structures, Spring 2024, Assignment 4

Instructor: Behrooz Mansouri (behrooz.mansouri@maine.edu)

Due: February 26, 2024

Notes for submission:

1. Starting from this assignment, students will work as a team, each team having 2 members. One student will act as the project manager, responsible for dividing the tasks and making the final submission. Students will work on GitHub, and each student will push their code to the repository using their own account. Grading is done based on the contribution to Git, and each student will be graded based on their task individually (i.e., you should not be concerned if your teammate has not finished their task and if their part of the code is not working properly).
2. Codes should be well-structured with comments to run. All the methods should contain a description of the return value, the inputs, and the goal of the method using JavaDoc
3. No method (including main) should be longer than 20 lines. If you have such methods, you should break them down into sub-methods. Note that as a good practice, a method should have a unique purpose
4. Any use of online resources should be explicitly indicated in the code

Objectives: This assignment is designed to review the concepts we learned about Queues.

Data: The data is similar to what we had in the previous assignment. The provided CSV (comma-separated values) file has the following columns, where each row represents one accident (total of 2.5M records):

ID, Severity, Start Time, End Time, Street, City, County, State, Temperature(F), Humidity(%), Visibility(mi), Weather Condition, Crossing, Sunrise/Sunset

Tasks:

In this assignment, you are going to estimate the counters needed in each county to finalize accident reports for a given county. Each county has a center with several counters; they will receive the reports and process them. Assume that all the counters will work 24/7. The time to process an accident report depends on the severity of the accident:

- a. 1: 60 mins

- b. 2: 120 mins
- c. 3: 180 mins
- d. 4: 240 mins

Follow these steps for your simulation:

1. For the input state and county, read the accident reports in an ArrayList, and then sorted based on the date of the start time (Ascending). All the accident reports should be processed on the same day they have happened (otherwise, you need more counters).
2. You will simulate this problem by iterating on each day (from the first to the last day). On each day, first, add all the accident reports that occurred on that day to the queue. Then, start processing each accident by removing them from the queue and then processing them (not an actual processing, just remove and consider the time required for processing).

Hint: Consider you have 5 counters. They will work for $5 \times 24 \times 60 = 7200$ minutes a day. So, in total, you have 7200 minutes to process your queue each day. Given this amount of time, start removing items from your queue, and based on the processing time, reduce the total amount of time, till it reaches zero. For example, if the first accident has a severity of 1, it will take 60 minutes. So the remaining time will be $7200 - 60$. But remember, all the reports should be processed within the same day as the accident. So when removing items from the queue, you should check how long they have been waiting and if it is more than a day, then you do not have enough counters.

3. For the following counties, calculate the minimum number of counters needed to process all the reports on time:
 - a. Los Angeles, CA
 - b. Orange, FL
 - c. Harris, TX
 - d. Hamilton, OH
 - e. New Castle, DE

Deliverables on GitHub:

1. All the Java classes should be in a package named "accidentpack"
2. All the necessary Java files with the README file for running the program. This file also contains a clear description of task division, the role of each team member, and the methods they have written.
3. There should be a program4.java file with a main method that takes in the CSV file path as the input argument and prints the answer to the following questions (by running your algorithm)
4. A PDF file showing your answers to the questions in step 3. Also for each sub-task, mention the complexity order of your algorithm and justify it
5. All the above items should be located on GitHub and the project manager will only submit the link to the Git repository. Make sure this is accessible

Expected run commands and outputs:

command: java program4 accidents.csv

outputs:

County: xx State: xx

xx seconds to simulate the process

minimum number of counters: xx