

Homework 4 Programming

(Maximum 25 points)
Due 11:59 pm Wednesday March 10, 2021

Submit your Java codes via Blackboard. Refer to the grading guidelines posted on Blackboard to understand how the submitted exercises will be graded.

1. (25) [Interval partitioning: programming] Implement the interval partitioning algorithm discussed in class. Your program should prompt the name of an input file containing a list of lectures (i.e., lecture number, start time, end time), read from the file, and then display output on terminal as specified below.

- (1) During the program run, output the classroom number and the start time and end time of each lecture allocated to a classroom.
- (2) At the end of the program run, output the list of all lectures allocated to each classroom.

Example input and output are shown below. The input is also provided in a text file "lectures1.txt".

Use a min-oriented priority queue of classroom nodes, where each node has two fields --- (i) a set of lectures scheduled in the classroom (Lecs) and (ii) the finish time of the last lecture scheduled in the classroom (lastFin); lastFin is used as the priority key. Feel free to use the Java priority queue class or write your own.

Submit all source codes as one program file. Your program codes should be working correctly and neatly organized and well commented.

Input in lectures1.txt (One lecture entry per line; entry format: (lecture number, start time, end time))

- (a, 1, 15)
- (b, 14, 18)
- (c, 2, 14)

Output for lectures in lectures1.txt (Two classrooms are allocated for the three lectures.)

During the program run:

- Classroom 1: (a, 1, 15)
- Classroom 2: (c, 2, 14)
- Classroom 2: (b, 14, 18)

At the end of the program run (*the order of classrooms in the final output does not matter*):

- Classroom 1: (a, 1, 15)
- Classroom 2: (c, 2, 14) (b, 14, 18)