

PROGRAMMING TASK(s)

Task 1 (100 points)

In this task, you are supposed to implement a Java program for the scheduling problem of a famous online order delivery company by using heap-based priority queue. The manager is trying to find how many couriers work in the order delivery part. For each courier in the order delivery, the expense of the company increases; but according to the standards of this company, the average waiting time for all customers should not exceed a given amount of time. So, the manager needs to optimize this number and calls for your help in this task. This company has the data of predict delivery time of customers. Your program should use these data to calculate average waiting times and find the minimum number of couriers needs to meet the average waiting time requirement. You must use your own implementation of heap-based priority queue by taking inspiration from your textbook or web. You are not allowed to use any external library or .jar file.

The input data are stored in a plain .txt file. The first line of the file shows the number of customers. The subsequent lines contain four integers, each separated by one or more whitespace characters (space or tab). These represent, respectively, the customer id, the registration year of the customer to the company (the company was founded in 1968), arrival time for order (in minutes from a given point, e.g. 12:00 am) and service time (in minutes). Here is the `sampleinput1.txt` file content:

```
12
1 1994 1 10
2 1974 1 14
3 2004 1 6
4 2004 1 5
5 1994 4 10
6 1974 7 14
7 1994 9 10
8 1974 11 14
9 2004 13 6
10 2004 14 5
11 1994 15 10
12 1974 17 14
```

For example, from the `sampleinput1.txt` file content above, it is seen that there are 12 customers. The first customer with id 1 gives order at minute 1, and his order is delivered in 10 minutes. He has been a customer of the company for 28 years (e.g. 2022 - 1994). The sixth customer with id 6 gives order at minute 7, and his order is delivered in 14 minutes. He has been a customer of the company for 48 years (2022 - 1974). The seventh customer with id 7 gives order at minute 9, and his order is delivered in 10 minutes. The first and seventh customers have a customer of the company for 28 years, which means that their priorities are the same.

You are asked to write a simulation program that reads customer data from `sampleinput1.txt` file and calculates the minimum number of couriers required for a given maximum average waiting time. The company assigns priority to its customers in order to provide them some special benefits. One of them is that they do not have to wait in the queue for order delivery service. The priority is defined considering the period of time that a customer has been with the company. Long-time customers of the company have higher priority than others. Your implementation must obey the following requirements:

- There may be at most 200 customers in the data file.
- The customer with the highest priority should be examined first.
- In case of having two customers with the same highest priority, the customer who has waited longer should be selected first.
- If more than one couriers is available at a given time; the customer is assigned to the courier with a lower id.
- When a courier starts giving a service to a customer, the courier should finish his service with this customer even though another customer with a higher priority gives order to company.
- Once a customer is assigned to a courier, the courier immediately starts delivering order of that customer and is not available during the service time given for that customer. After the service of that customer carries out, the courier becomes available immediately.
- The waiting time of a customer is the duration (difference) between the given order time of the customer and the time he is assigned to a courier.

In your implementation, you must use a heap-based priority queue to store customers who are waiting for a courier (i.e., to store customers who have gave order to their order have not been delivered yet). If you do not use such a heap-based priority queue to store these customers, then you will get no points from this task of homework.

Firstly, you should read input txt file from the user. Then, you should read the maximum average waiting time. Your program should calculate the minimum number of courier required for meeting this maximum average waiting time. Please check your program with this input file as well as the others that you will create. Please note that we will use other input files when grading your assignments. Sample run is shown as follows, green texts represent user inputs:

```
Enter input filename:
sampleinput1.txt
Enter the maximum average waiting time:
10

Minimum number of couriers required: 3

Simulation with 3 Couriers:

Courier 0 takes customer 2 at minute 1 (wait: 0 mins)
Courier 1 takes customer 1 at minute 1 (wait: 0 mins)
Courier 2 takes customer 3 at minute 1 (wait: 0 mins)
Courier 2 takes customer 6 at minute 7 (wait: 0 mins)
Courier 1 takes customer 8 at minute 11 (wait: 0 mins)
Courier 0 takes customer 5 at minute 15 (wait: 11 mins)
Courier 2 takes customer 12 at minute 21 (wait: 4 mins)
Courier 0 takes customer 7 at minute 25 (wait: 16 mins)
Courier 1 takes customer 11 at minute 25 (wait: 10 mins)
Courier 0 takes customer 4 at minute 35 (wait: 34 mins)
Courier 1 takes customer 9 at minute 35 (wait: 22 mins)
Courier 2 takes customer 10 at minute 35 (wait: 21 mins)

Average waiting time: 9.83333 minutes
```

As a hint, you use the heap data structure to hold customers that are waiting for a courier and to find the customer with the highest priority. Update the heap whenever a new customer gives order or a customer's order is delivered. In order to find the optimum number of couriers needed, repeat the simulation for increasing number of couriers and return the minimum number of waiters that will achieve the maximum average waiting time constraint. Display the simulation for which you find the optimum number of couriers.