

**QUEUES SIMULATOR**

Student: Francesca Dițulescu

Teacher: Viorica Chifu

Table of contents

Assignment objective . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3

Main objective . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .3

Sub-objectives . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .3

Problem analysis, modeling, scenarios, use cases . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .3

Problem analysis . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .3

Solution . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .3

Requirements . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3

Use cases . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .4

Design . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .5

Overall System Design . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .5

Division into packages . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .6

Division into classes . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .8

UML Class Diagram . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 7

GUI Design . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 7

Implementation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8

Results . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .12

Conclusions . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 31

Bibliography . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 31

ASSIGNMENT OBJECTIVE

MAIN OBJECTIVE

The main objective of this project is to design and implement a queues simulator manager with a dedicated graphical user interface in order to analyze queuing based systems for determining and minimizing clients’ waiting time.

SUB-OBJECTIVES

* Analyze the problem and identify the requirements
* Design the queues simulation manager
* Implement the queues simulation manager
* Test the queues simulation manager

PROBLEM ANALYSIS

PROBLEM

I think this project comes into helping find a solution to the following problem: it is meant to help owers from stores how can they deal with a great amount of customers while having a specific number of cash registers available. It also comes into providing a great understanding of how multiple queues should work considering the time interval the store is open.

SOLUTION

As a solution, this projects provides a faster and more interactive application that can help the user see how queues can be managed in a place.

REQUIREMENTS

Functional requirements

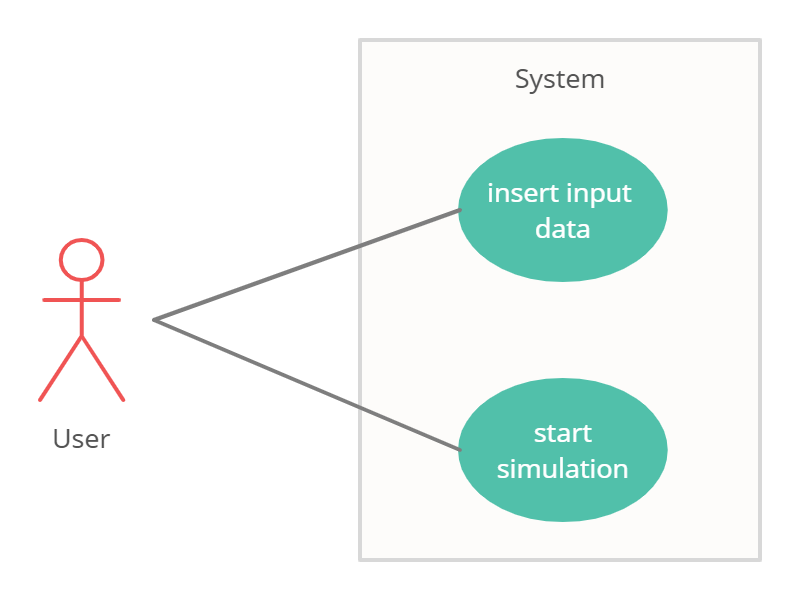
* The queues simulator should allow the user to insert data.
* The queues simulator should allow the user to start the simulation (press the start button).
* The queues simulator should be able to print, at each second, the state of the simulation on the graphical user interface.
* The queues simulator should be able to print, for each second, the state of the simulation in a .txt file.
* The queues simulator should be able to run for as long as the simulation time introduced.

Non-functional requirements

* The queues simulator should be intuitive and easy to use by the user.
* The queues simulator should have a nice and pleasant graphical user interface.

USE CASES

The Use Case Diagram:



1. Insert input data

Primary actor: user

Main success scenario:

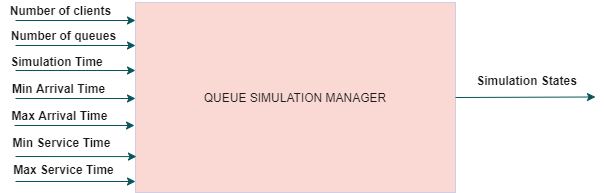
1. The user introduces valid input data in the graphical user interface.
2. The user selects the „Start simulation” button.
3. The queues simulation manager performs the simulation and runs as long as the running time introduced by the user.

Alternative sequence:

1. The user introduces invalid input data in the graphical user interface:
   1. Invalid number of clients.
   2. Invalid number of queues.
   3. Invalid simulation running time.
   4. Invalid bounds for the arrival time.
   5. Invalid bounds for the service time.
2. The scenario returns to step 1.

DESIGN

Level 1: Overall System Design



Level 2: Division into sub-systems/packages

MVC Architecture

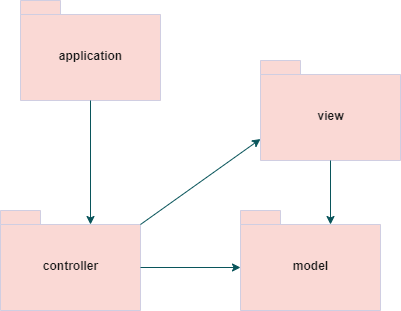
The Model-View-Controller (MVC) is an architectural pattern that separates an application into three main logical components: the model, the view, and the controller. Each of these components are built to handle specific development aspects of an application. MVC is one of the most frequently used industry-standard web development framework to create scalable and extensible projects.

The Model component corresponds to all the data-related logic that the user works with. This can represent either the data that is being transferred between the View and Controller components or any other business logic-related data.

The View component is used for all the UI logic of the application.

Controllers act as an interface between Model and View components to process all the business logic and incoming requests, manipulate data using the Model component and interact with the Views to render the final output.

My package diagram is the following:



Level 3: Division into classes

a) APPLICATION PACKAGE – contains the main function (named SimulationManager) which runs the application itself. It contains:

- SimulationManager class – it is meant to represent the whole ”environment” where the action (all the queues and all the clients coming) is happening.

- FileWriter class – it deasl with writing the log of events in the .txt file.

b) MODEL PACKAGE – contains the classes which model the application (what we concretely use in order to have the application):

- Client class – represents the clients that can be added to the queue and that needs to be served. It is described by an ID, arrival time (when does the client arrive to be served) and service time (how long it takes to be served). During the implementation of the project, another field (waiting time) had to be added in order to see how much time does the client waits in the queue until it is served.

- Server class – represents the queue formed. It is composed of a list of clients and is described by an atomic integer which computes the processing time of the queue (how long does it take for the whole queue to get empty). In addition, two more fields (totalWaitingTime and numberOfClientsServed) have been added to the class for the purpose of computing the analytics after the simulation.

c) CONTROLLER PACKAGE – deals with the logic behind that makes the application work properly. It contains 3 classes and one interface:

- Controller – deals with how the application should perfom based on the buttons pressed by the user.

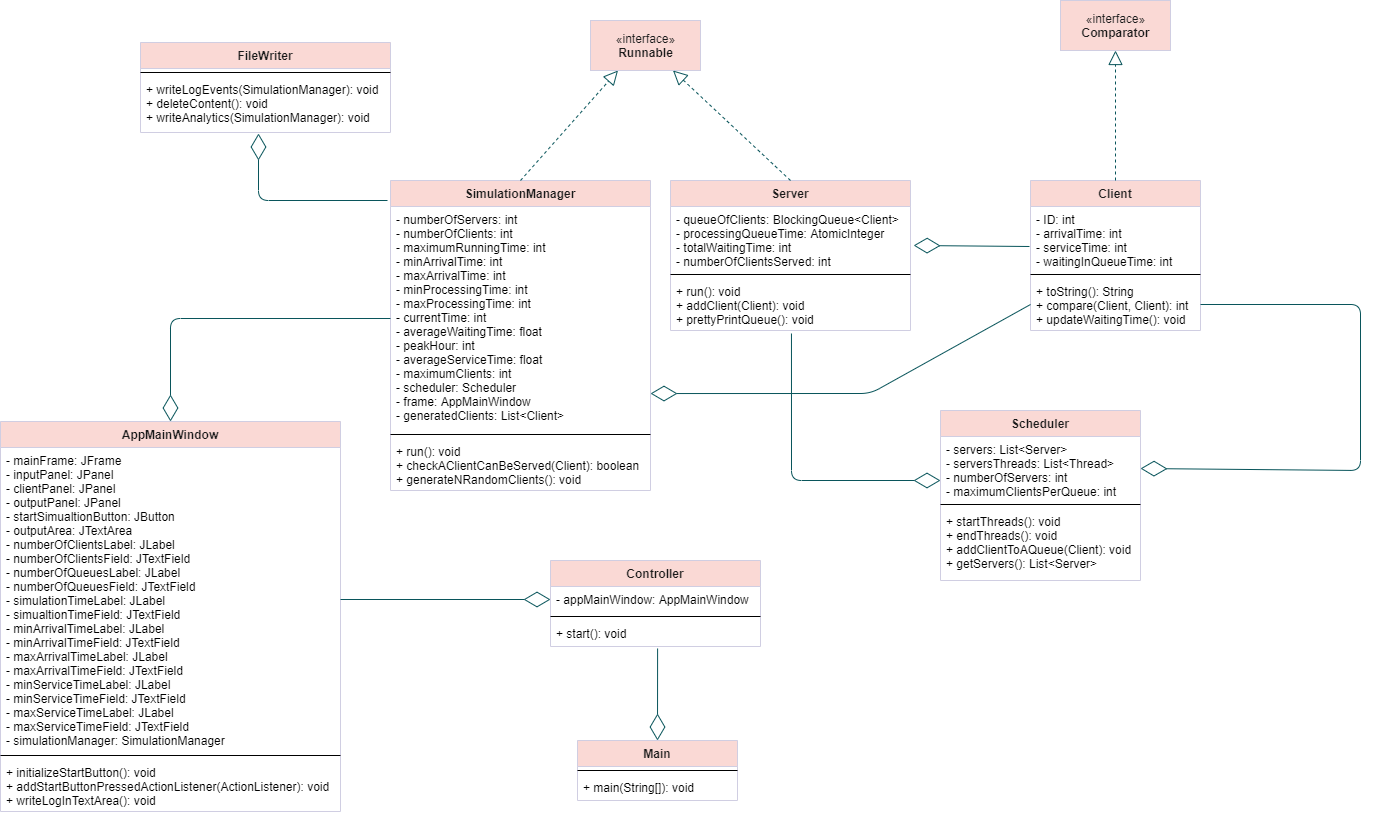
- Scheduler – another controller used to deal with specific operations during the simulation (adding a client, accessing the servers list).

- SimulationManager – class meant to represent the whole environment where the simulation is going. It deals with the flow of actions (when to add a client, what to modify in the meatime and takes into account the parameters

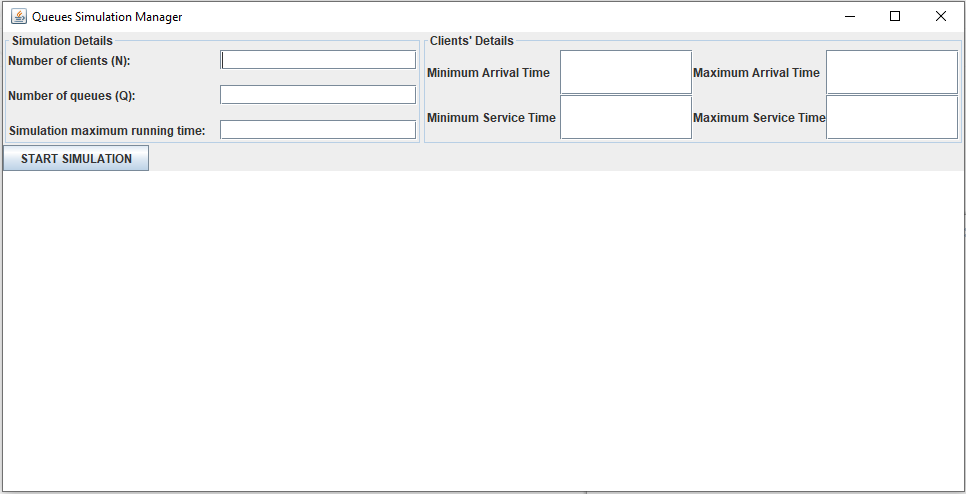
d) VIEW PACKAGE – contains the class that creates the graphical user interface (all the buttons, frames, panels, etc):

- AppMainWindow class – composed of one main frame and three panels. It creates the graphical user interface.

UML CLASS DIAGRAM



GUI DESIGN



There is only one main frame with three panels. Each panel is divided by its scope: the first panel is the Simulation Details one where the user has to introduce input data regarding the simulation, the second panel is the Clients’ Details one where the user has to introduce input data regarding the clients’ arrival and service times. The third and the last panel is the one with the START button and the text area used to show in real time the simulation’s states.

IMPLEMENTATION

DATA STRUCTURES USED

For this assignment I have used the LinkedBlockingQueue data structure which implements the BlockingQueue interface. Its utility for this application is for synchronizing threads and protecting the critical zones from being accessed by more than one thread by saving all the upcoming threads in a queue and dequeue a thread when the critical zone is available. If the critical zones are not available (are still accessed by another thread), all the threads that will try to access those zones will be queued in the LinkedBlockingQueue structure and kept there until the next thread can go and execute its task.

In addition, I have used mostly ArrayList in order to store my servers list (all the queues) and the clients that were generated with random data for the simulation.

MULTITHREADING

Multithreading is the ability of a [program](https://searchsoftwarequality.techtarget.com/definition/program) or an [operating system](https://whatis.techtarget.com/definition/operating-system-OS) [process](https://whatis.techtarget.com/definition/process) to manage its use by more than one user at a time and to even manage multiple requests by the same user without having to have multiple copies of the programming running in the computer. Each user request for a program or system service (and here a user can also be another program) is kept track of as a [thread](https://whatis.techtarget.com/definition/thread) with a separate identity. As programs work on behalf of the initial request for that thread and are interrupted by other requests, the status of work on behalf of that thread is kept track of until the work is completed.

I will describe now how I thought when implementing the run methods for the two classes that implemented the Runnable interface.

For the server threads (the queues), the approach was the following:

Just like in a store, there are multiple cash registers which are available and have clients waiting in the line at the same time. Each client takes some time (in our case seconds) to be served and to finish everything (collecting all the items, paying and putting everything in the bag).

For each queue, the action is executed in the same manner:

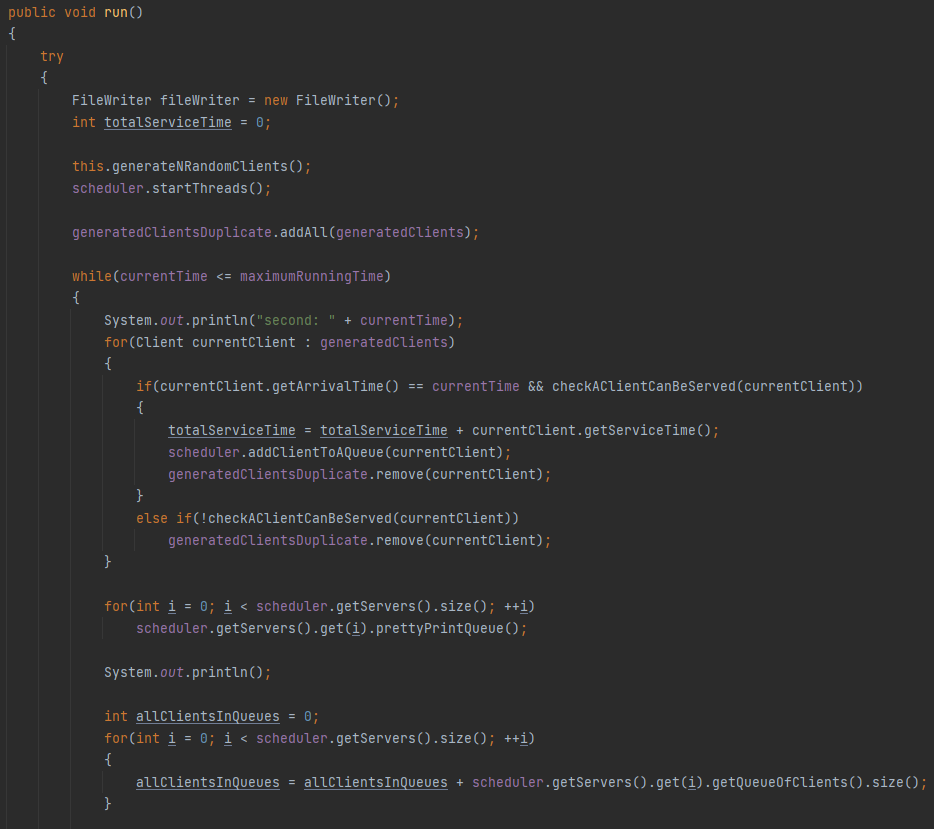
* + Take the first client from the queue
  + Let him do its job for serviceTime seconds (let the thread wait for that much to simulate the time needed by the client to do all the things)
  + After the client is done, it is removed from the queue and goes on with his other stuff (which does not interest us).

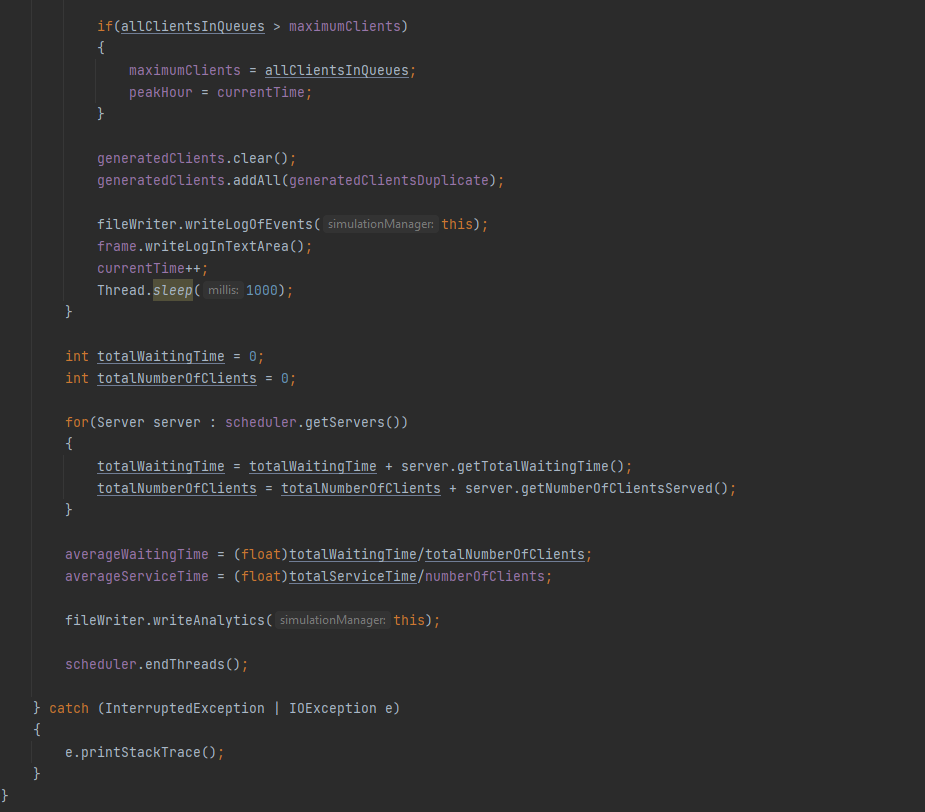
In this manner, the thread function of the server is focused on taking with the peek method the first element of the queue of clients, sleep for serviceTime seconds to simulate the client’s time and, in the end, to remove it from the queue when the thread is re-awakened.

For the simulation manager thread (the main one), the approach was the following:

In order to simulate a real-life situation, all the threads (queues) had to be running at the same time (multithreading). This was achieved with the help of the Scheduler (which kind of works like a controller) using the method to start all threads at the same time. Having them active at the same time, the SimulationManager was able to send each client to the best queue according to our criteria (although people in real life do not choose queues this way). It was always focusing on all the statuses of the clients and was only put on sleep for one second such that it simulates a real time running period.

SIMULATION MANAGER RUN METHOD





SERVER RUN METHOD



RESULTS

FIRST TEST

Time: 0  
Waiting clients: Client(1, 19, 2); Client(2, 21, 2); Client(3, 24, 3); Client(4, 26, 4);   
Queue 1: closed  
Queue 2: closed  
  
Time: 1  
Waiting clients: Client(1, 19, 2); Client(2, 21, 2); Client(3, 24, 3); Client(4, 26, 4);   
Queue 1: closed  
Queue 2: closed  
  
Time: 2  
Waiting clients: Client(1, 19, 2); Client(2, 21, 2); Client(3, 24, 3); Client(4, 26, 4);   
Queue 1: closed  
Queue 2: closed  
  
Time: 3  
Waiting clients: Client(1, 19, 2); Client(2, 21, 2); Client(3, 24, 3); Client(4, 26, 4);   
Queue 1: closed  
Queue 2: closed  
  
Time: 4  
Waiting clients: Client(1, 19, 2); Client(2, 21, 2); Client(3, 24, 3); Client(4, 26, 4);   
Queue 1: closed  
Queue 2: closed  
  
Time: 5  
Waiting clients: Client(1, 19, 2); Client(2, 21, 2); Client(3, 24, 3); Client(4, 26, 4);   
Queue 1: closed  
Queue 2: closed  
  
Time: 6  
Waiting clients: Client(1, 19, 2); Client(2, 21, 2); Client(3, 24, 3); Client(4, 26, 4);   
Queue 1: closed  
Queue 2: closed  
  
Time: 7  
Waiting clients: Client(1, 19, 2); Client(2, 21, 2); Client(3, 24, 3); Client(4, 26, 4);   
Queue 1: closed  
Queue 2: closed  
  
Time: 8  
Waiting clients: Client(1, 19, 2); Client(2, 21, 2); Client(3, 24, 3); Client(4, 26, 4);   
Queue 1: closed  
Queue 2: closed  
  
Time: 9  
Waiting clients: Client(1, 19, 2); Client(2, 21, 2); Client(3, 24, 3); Client(4, 26, 4);   
Queue 1: closed  
Queue 2: closed  
  
Time: 10  
Waiting clients: Client(1, 19, 2); Client(2, 21, 2); Client(3, 24, 3); Client(4, 26, 4);   
Queue 1: closed  
Queue 2: closed  
  
Time: 11  
Waiting clients: Client(1, 19, 2); Client(2, 21, 2); Client(3, 24, 3); Client(4, 26, 4);   
Queue 1: closed  
Queue 2: closed  
  
Time: 12  
Waiting clients: Client(1, 19, 2); Client(2, 21, 2); Client(3, 24, 3); Client(4, 26, 4);   
Queue 1: closed  
Queue 2: closed  
  
Time: 13  
Waiting clients: Client(1, 19, 2); Client(2, 21, 2); Client(3, 24, 3); Client(4, 26, 4);   
Queue 1: closed  
Queue 2: closed  
  
Time: 14  
Waiting clients: Client(1, 19, 2); Client(2, 21, 2); Client(3, 24, 3); Client(4, 26, 4);   
Queue 1: closed  
Queue 2: closed  
  
Time: 15  
Waiting clients: Client(1, 19, 2); Client(2, 21, 2); Client(3, 24, 3); Client(4, 26, 4);   
Queue 1: closed  
Queue 2: closed  
  
Time: 16  
Waiting clients: Client(1, 19, 2); Client(2, 21, 2); Client(3, 24, 3); Client(4, 26, 4);   
Queue 1: closed  
Queue 2: closed  
  
Time: 17  
Waiting clients: Client(1, 19, 2); Client(2, 21, 2); Client(3, 24, 3); Client(4, 26, 4);   
Queue 1: closed  
Queue 2: closed  
  
Time: 18  
Waiting clients: Client(1, 19, 2); Client(2, 21, 2); Client(3, 24, 3); Client(4, 26, 4);   
Queue 1: closed  
Queue 2: closed  
  
Time: 19  
Waiting clients: Client(2, 21, 2); Client(3, 24, 3); Client(4, 26, 4);   
Queue 1: Client(1, 19, 2);   
Queue 2: closed  
  
Time: 20  
Waiting clients: Client(2, 21, 2); Client(3, 24, 3); Client(4, 26, 4);   
Queue 1: Client(1, 19, 1);   
Queue 2: closed  
  
Time: 21  
Waiting clients: Client(3, 24, 3); Client(4, 26, 4);   
Queue 1: closed  
Queue 2: Client(2, 21, 2);   
  
Time: 22  
Waiting clients: Client(3, 24, 3); Client(4, 26, 4);   
Queue 1: closed  
Queue 2: Client(2, 21, 1);   
  
Time: 23  
Waiting clients: Client(3, 24, 3); Client(4, 26, 4);   
Queue 1: closed  
Queue 2: closed  
  
Time: 24  
Waiting clients: Client(4, 26, 4);   
Queue 1: Client(3, 24, 3);   
Queue 2: closed  
  
Time: 25  
Waiting clients: Client(4, 26, 4);   
Queue 1: Client(3, 24, 2);   
Queue 2: closed  
  
Time: 26  
Waiting clients:   
Queue 1: Client(3, 24, 1);   
Queue 2: Client(4, 26, 4);   
  
Time: 27  
Waiting clients:   
Queue 1: closed  
Queue 2: Client(4, 26, 3);   
  
Time: 28  
Waiting clients:   
Queue 1: closed  
Queue 2: Client(4, 26, 2);   
  
Time: 29  
Waiting clients:   
Queue 1: closed  
Queue 2: Client(4, 26, 1);   
  
Time: 30  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
  
Time: 31  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
  
Time: 32  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
  
Time: 33  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
  
Time: 34  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
  
Time: 35  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
  
Time: 36  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
  
Time: 37  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
  
Time: 38  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
  
Time: 39  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
  
Time: 40  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
  
Time: 41  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
  
Time: 42  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
  
Time: 43  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
  
Time: 44  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
  
Time: 45  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
  
Time: 46  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
  
Time: 47  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
  
Time: 48  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
  
Time: 49  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
  
Time: 50  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
  
Time: 51  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
  
Time: 52  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
  
Time: 53  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
  
Time: 54  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
  
Time: 55  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
  
Time: 56  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
  
Time: 57  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
  
Time: 58  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
  
Time: 59  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
  
Time: 60  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
  
Average Waiting Time: 0.0  
Average Service Time: 2.75  
Peak Hour: 26

SECOND TEST

Time: 0  
Waiting clients: Client(1, 2, 3); Client(2, 3, 2); Client(3, 4, 6); Client(4, 5, 2); Client(5, 5, 5); Client(6, 6, 2); Client(7, 7, 2); Client(8, 7, 3); Client(9, 7, 4); Client(10, 7, 2); Client(11, 8, 3); Client(12, 10, 7); Client(13, 10, 2); Client(14, 11, 1); Client(15, 12, 5); Client(16, 12, 6); Client(17, 14, 5); Client(18, 15, 4); Client(19, 19, 1); Client(20, 19, 6); Client(21, 20, 4); Client(22, 21, 6); Client(23, 22, 5); Client(24, 22, 5); Client(25, 23, 7); Client(26, 23, 2); Client(27, 24, 7); Client(28, 24, 7); Client(29, 25, 1); Client(30, 26, 1); Client(31, 26, 6); Client(32, 28, 4); Client(33, 29, 6); Client(34, 29, 3); Client(35, 29, 4); Client(36, 29, 6); Client(37, 29, 4); Client(38, 30, 1); Client(39, 32, 4); Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: closed  
Queue 2: closed  
Queue 3: closed  
Queue 4: closed  
Queue 5: closed  
  
Time: 1  
Waiting clients: Client(1, 2, 3); Client(2, 3, 2); Client(3, 4, 6); Client(4, 5, 2); Client(5, 5, 5); Client(6, 6, 2); Client(7, 7, 2); Client(8, 7, 3); Client(9, 7, 4); Client(10, 7, 2); Client(11, 8, 3); Client(12, 10, 7); Client(13, 10, 2); Client(14, 11, 1); Client(15, 12, 5); Client(16, 12, 6); Client(17, 14, 5); Client(18, 15, 4); Client(19, 19, 1); Client(20, 19, 6); Client(21, 20, 4); Client(22, 21, 6); Client(23, 22, 5); Client(24, 22, 5); Client(25, 23, 7); Client(26, 23, 2); Client(27, 24, 7); Client(28, 24, 7); Client(29, 25, 1); Client(30, 26, 1); Client(31, 26, 6); Client(32, 28, 4); Client(33, 29, 6); Client(34, 29, 3); Client(35, 29, 4); Client(36, 29, 6); Client(37, 29, 4); Client(38, 30, 1); Client(39, 32, 4); Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: closed  
Queue 2: closed  
Queue 3: closed  
Queue 4: closed  
Queue 5: closed  
  
Time: 2  
Waiting clients: Client(2, 3, 2); Client(3, 4, 6); Client(4, 5, 2); Client(5, 5, 5); Client(6, 6, 2); Client(7, 7, 2); Client(8, 7, 3); Client(9, 7, 4); Client(10, 7, 2); Client(11, 8, 3); Client(12, 10, 7); Client(13, 10, 2); Client(14, 11, 1); Client(15, 12, 5); Client(16, 12, 6); Client(17, 14, 5); Client(18, 15, 4); Client(19, 19, 1); Client(20, 19, 6); Client(21, 20, 4); Client(22, 21, 6); Client(23, 22, 5); Client(24, 22, 5); Client(25, 23, 7); Client(26, 23, 2); Client(27, 24, 7); Client(28, 24, 7); Client(29, 25, 1); Client(30, 26, 1); Client(31, 26, 6); Client(32, 28, 4); Client(33, 29, 6); Client(34, 29, 3); Client(35, 29, 4); Client(36, 29, 6); Client(37, 29, 4); Client(38, 30, 1); Client(39, 32, 4); Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: Client(1, 2, 3);   
Queue 2: closed  
Queue 3: closed  
Queue 4: closed  
Queue 5: closed  
  
Time: 3  
Waiting clients: Client(3, 4, 6); Client(4, 5, 2); Client(5, 5, 5); Client(6, 6, 2); Client(7, 7, 2); Client(8, 7, 3); Client(9, 7, 4); Client(10, 7, 2); Client(11, 8, 3); Client(12, 10, 7); Client(13, 10, 2); Client(14, 11, 1); Client(15, 12, 5); Client(16, 12, 6); Client(17, 14, 5); Client(18, 15, 4); Client(19, 19, 1); Client(20, 19, 6); Client(21, 20, 4); Client(22, 21, 6); Client(23, 22, 5); Client(24, 22, 5); Client(25, 23, 7); Client(26, 23, 2); Client(27, 24, 7); Client(28, 24, 7); Client(29, 25, 1); Client(30, 26, 1); Client(31, 26, 6); Client(32, 28, 4); Client(33, 29, 6); Client(34, 29, 3); Client(35, 29, 4); Client(36, 29, 6); Client(37, 29, 4); Client(38, 30, 1); Client(39, 32, 4); Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: Client(1, 2, 2);   
Queue 2: Client(2, 3, 2);   
Queue 3: closed  
Queue 4: closed  
Queue 5: closed  
  
Time: 4  
Waiting clients: Client(4, 5, 2); Client(5, 5, 5); Client(6, 6, 2); Client(7, 7, 2); Client(8, 7, 3); Client(9, 7, 4); Client(10, 7, 2); Client(11, 8, 3); Client(12, 10, 7); Client(13, 10, 2); Client(14, 11, 1); Client(15, 12, 5); Client(16, 12, 6); Client(17, 14, 5); Client(18, 15, 4); Client(19, 19, 1); Client(20, 19, 6); Client(21, 20, 4); Client(22, 21, 6); Client(23, 22, 5); Client(24, 22, 5); Client(25, 23, 7); Client(26, 23, 2); Client(27, 24, 7); Client(28, 24, 7); Client(29, 25, 1); Client(30, 26, 1); Client(31, 26, 6); Client(32, 28, 4); Client(33, 29, 6); Client(34, 29, 3); Client(35, 29, 4); Client(36, 29, 6); Client(37, 29, 4); Client(38, 30, 1); Client(39, 32, 4); Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: Client(1, 2, 1);   
Queue 2: Client(2, 3, 1);   
Queue 3: Client(3, 4, 6);   
Queue 4: closed  
Queue 5: closed  
  
Time: 5  
Waiting clients: Client(6, 6, 2); Client(7, 7, 2); Client(8, 7, 3); Client(9, 7, 4); Client(10, 7, 2); Client(11, 8, 3); Client(12, 10, 7); Client(13, 10, 2); Client(14, 11, 1); Client(15, 12, 5); Client(16, 12, 6); Client(17, 14, 5); Client(18, 15, 4); Client(19, 19, 1); Client(20, 19, 6); Client(21, 20, 4); Client(22, 21, 6); Client(23, 22, 5); Client(24, 22, 5); Client(25, 23, 7); Client(26, 23, 2); Client(27, 24, 7); Client(28, 24, 7); Client(29, 25, 1); Client(30, 26, 1); Client(31, 26, 6); Client(32, 28, 4); Client(33, 29, 6); Client(34, 29, 3); Client(35, 29, 4); Client(36, 29, 6); Client(37, 29, 4); Client(38, 30, 1); Client(39, 32, 4); Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: closed  
Queue 2: closed  
Queue 3: Client(3, 4, 5);   
Queue 4: Client(4, 5, 2);   
Queue 5: Client(5, 5, 5);   
  
Time: 6  
Waiting clients: Client(7, 7, 2); Client(8, 7, 3); Client(9, 7, 4); Client(10, 7, 2); Client(11, 8, 3); Client(12, 10, 7); Client(13, 10, 2); Client(14, 11, 1); Client(15, 12, 5); Client(16, 12, 6); Client(17, 14, 5); Client(18, 15, 4); Client(19, 19, 1); Client(20, 19, 6); Client(21, 20, 4); Client(22, 21, 6); Client(23, 22, 5); Client(24, 22, 5); Client(25, 23, 7); Client(26, 23, 2); Client(27, 24, 7); Client(28, 24, 7); Client(29, 25, 1); Client(30, 26, 1); Client(31, 26, 6); Client(32, 28, 4); Client(33, 29, 6); Client(34, 29, 3); Client(35, 29, 4); Client(36, 29, 6); Client(37, 29, 4); Client(38, 30, 1); Client(39, 32, 4); Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: closed  
Queue 2: Client(6, 6, 2);   
Queue 3: Client(3, 4, 4);   
Queue 4: Client(4, 5, 1);   
Queue 5: Client(5, 5, 4);   
  
Time: 7  
Waiting clients: Client(11, 8, 3); Client(12, 10, 7); Client(13, 10, 2); Client(14, 11, 1); Client(15, 12, 5); Client(16, 12, 6); Client(17, 14, 5); Client(18, 15, 4); Client(19, 19, 1); Client(20, 19, 6); Client(21, 20, 4); Client(22, 21, 6); Client(23, 22, 5); Client(24, 22, 5); Client(25, 23, 7); Client(26, 23, 2); Client(27, 24, 7); Client(28, 24, 7); Client(29, 25, 1); Client(30, 26, 1); Client(31, 26, 6); Client(32, 28, 4); Client(33, 29, 6); Client(34, 29, 3); Client(35, 29, 4); Client(36, 29, 6); Client(37, 29, 4); Client(38, 30, 1); Client(39, 32, 4); Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: Client(8, 7, 3);   
Queue 2: Client(6, 6, 1); Client(9, 7, 4);   
Queue 3: Client(3, 4, 3);   
Queue 4: Client(7, 7, 2); Client(10, 7, 2);   
Queue 5: Client(5, 5, 3);   
  
Time: 8  
Waiting clients: Client(12, 10, 7); Client(13, 10, 2); Client(14, 11, 1); Client(15, 12, 5); Client(16, 12, 6); Client(17, 14, 5); Client(18, 15, 4); Client(19, 19, 1); Client(20, 19, 6); Client(21, 20, 4); Client(22, 21, 6); Client(23, 22, 5); Client(24, 22, 5); Client(25, 23, 7); Client(26, 23, 2); Client(27, 24, 7); Client(28, 24, 7); Client(29, 25, 1); Client(30, 26, 1); Client(31, 26, 6); Client(32, 28, 4); Client(33, 29, 6); Client(34, 29, 3); Client(35, 29, 4); Client(36, 29, 6); Client(37, 29, 4); Client(38, 30, 1); Client(39, 32, 4); Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: Client(8, 7, 2);   
Queue 2: Client(9, 7, 4);   
Queue 3: Client(3, 4, 2);   
Queue 4: Client(7, 7, 1); Client(10, 7, 2);   
Queue 5: Client(5, 5, 2); Client(11, 8, 3);   
  
Time: 9  
Waiting clients: Client(12, 10, 7); Client(13, 10, 2); Client(14, 11, 1); Client(15, 12, 5); Client(16, 12, 6); Client(17, 14, 5); Client(18, 15, 4); Client(19, 19, 1); Client(20, 19, 6); Client(21, 20, 4); Client(22, 21, 6); Client(23, 22, 5); Client(24, 22, 5); Client(25, 23, 7); Client(26, 23, 2); Client(27, 24, 7); Client(28, 24, 7); Client(29, 25, 1); Client(30, 26, 1); Client(31, 26, 6); Client(32, 28, 4); Client(33, 29, 6); Client(34, 29, 3); Client(35, 29, 4); Client(36, 29, 6); Client(37, 29, 4); Client(38, 30, 1); Client(39, 32, 4); Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: Client(8, 7, 1);   
Queue 2: Client(9, 7, 3);   
Queue 3: Client(3, 4, 1);   
Queue 4: Client(10, 7, 2);   
Queue 5: Client(5, 5, 1); Client(11, 8, 3);   
  
Time: 10  
Waiting clients: Client(14, 11, 1); Client(15, 12, 5); Client(16, 12, 6); Client(17, 14, 5); Client(18, 15, 4); Client(19, 19, 1); Client(20, 19, 6); Client(21, 20, 4); Client(22, 21, 6); Client(23, 22, 5); Client(24, 22, 5); Client(25, 23, 7); Client(26, 23, 2); Client(27, 24, 7); Client(28, 24, 7); Client(29, 25, 1); Client(30, 26, 1); Client(31, 26, 6); Client(32, 28, 4); Client(33, 29, 6); Client(34, 29, 3); Client(35, 29, 4); Client(36, 29, 6); Client(37, 29, 4); Client(38, 30, 1); Client(39, 32, 4); Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: Client(12, 10, 7);   
Queue 2: Client(9, 7, 2);   
Queue 3: Client(13, 10, 2);   
Queue 4: Client(10, 7, 1);   
Queue 5: Client(11, 8, 3);   
  
Time: 11  
Waiting clients: Client(15, 12, 5); Client(16, 12, 6); Client(17, 14, 5); Client(18, 15, 4); Client(19, 19, 1); Client(20, 19, 6); Client(21, 20, 4); Client(22, 21, 6); Client(23, 22, 5); Client(24, 22, 5); Client(25, 23, 7); Client(26, 23, 2); Client(27, 24, 7); Client(28, 24, 7); Client(29, 25, 1); Client(30, 26, 1); Client(31, 26, 6); Client(32, 28, 4); Client(33, 29, 6); Client(34, 29, 3); Client(35, 29, 4); Client(36, 29, 6); Client(37, 29, 4); Client(38, 30, 1); Client(39, 32, 4); Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: Client(12, 10, 6);   
Queue 2: Client(9, 7, 1);   
Queue 3: Client(13, 10, 1);   
Queue 4: Client(14, 11, 1);   
Queue 5: Client(11, 8, 2);   
  
Time: 12  
Waiting clients: Client(17, 14, 5); Client(18, 15, 4); Client(19, 19, 1); Client(20, 19, 6); Client(21, 20, 4); Client(22, 21, 6); Client(23, 22, 5); Client(24, 22, 5); Client(25, 23, 7); Client(26, 23, 2); Client(27, 24, 7); Client(28, 24, 7); Client(29, 25, 1); Client(30, 26, 1); Client(31, 26, 6); Client(32, 28, 4); Client(33, 29, 6); Client(34, 29, 3); Client(35, 29, 4); Client(36, 29, 6); Client(37, 29, 4); Client(38, 30, 1); Client(39, 32, 4); Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: Client(12, 10, 5);   
Queue 2: Client(16, 12, 6);   
Queue 3: closed  
Queue 4: Client(15, 12, 5);   
Queue 5: Client(11, 8, 1);   
  
Time: 13  
Waiting clients: Client(17, 14, 5); Client(18, 15, 4); Client(19, 19, 1); Client(20, 19, 6); Client(21, 20, 4); Client(22, 21, 6); Client(23, 22, 5); Client(24, 22, 5); Client(25, 23, 7); Client(26, 23, 2); Client(27, 24, 7); Client(28, 24, 7); Client(29, 25, 1); Client(30, 26, 1); Client(31, 26, 6); Client(32, 28, 4); Client(33, 29, 6); Client(34, 29, 3); Client(35, 29, 4); Client(36, 29, 6); Client(37, 29, 4); Client(38, 30, 1); Client(39, 32, 4); Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: Client(12, 10, 4);   
Queue 2: Client(16, 12, 5);   
Queue 3: closed  
Queue 4: Client(15, 12, 4);   
Queue 5: closed  
  
Time: 14  
Waiting clients: Client(18, 15, 4); Client(19, 19, 1); Client(20, 19, 6); Client(21, 20, 4); Client(22, 21, 6); Client(23, 22, 5); Client(24, 22, 5); Client(25, 23, 7); Client(26, 23, 2); Client(27, 24, 7); Client(28, 24, 7); Client(29, 25, 1); Client(30, 26, 1); Client(31, 26, 6); Client(32, 28, 4); Client(33, 29, 6); Client(34, 29, 3); Client(35, 29, 4); Client(36, 29, 6); Client(37, 29, 4); Client(38, 30, 1); Client(39, 32, 4); Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: Client(12, 10, 3);   
Queue 2: Client(16, 12, 4);   
Queue 3: Client(17, 14, 5);   
Queue 4: Client(15, 12, 3);   
Queue 5: closed  
  
Time: 15  
Waiting clients: Client(19, 19, 1); Client(20, 19, 6); Client(21, 20, 4); Client(22, 21, 6); Client(23, 22, 5); Client(24, 22, 5); Client(25, 23, 7); Client(26, 23, 2); Client(27, 24, 7); Client(28, 24, 7); Client(29, 25, 1); Client(30, 26, 1); Client(31, 26, 6); Client(32, 28, 4); Client(33, 29, 6); Client(34, 29, 3); Client(35, 29, 4); Client(36, 29, 6); Client(37, 29, 4); Client(38, 30, 1); Client(39, 32, 4); Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: Client(12, 10, 2);   
Queue 2: Client(16, 12, 3);   
Queue 3: Client(17, 14, 4);   
Queue 4: Client(15, 12, 2);   
Queue 5: Client(18, 15, 4);   
  
Time: 16  
Waiting clients: Client(19, 19, 1); Client(20, 19, 6); Client(21, 20, 4); Client(22, 21, 6); Client(23, 22, 5); Client(24, 22, 5); Client(25, 23, 7); Client(26, 23, 2); Client(27, 24, 7); Client(28, 24, 7); Client(29, 25, 1); Client(30, 26, 1); Client(31, 26, 6); Client(32, 28, 4); Client(33, 29, 6); Client(34, 29, 3); Client(35, 29, 4); Client(36, 29, 6); Client(37, 29, 4); Client(38, 30, 1); Client(39, 32, 4); Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: Client(12, 10, 1);   
Queue 2: Client(16, 12, 2);   
Queue 3: Client(17, 14, 3);   
Queue 4: Client(15, 12, 1);   
Queue 5: Client(18, 15, 3);   
  
Time: 17  
Waiting clients: Client(19, 19, 1); Client(20, 19, 6); Client(21, 20, 4); Client(22, 21, 6); Client(23, 22, 5); Client(24, 22, 5); Client(25, 23, 7); Client(26, 23, 2); Client(27, 24, 7); Client(28, 24, 7); Client(29, 25, 1); Client(30, 26, 1); Client(31, 26, 6); Client(32, 28, 4); Client(33, 29, 6); Client(34, 29, 3); Client(35, 29, 4); Client(36, 29, 6); Client(37, 29, 4); Client(38, 30, 1); Client(39, 32, 4); Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: closed  
Queue 2: Client(16, 12, 1);   
Queue 3: Client(17, 14, 2);   
Queue 4: closed  
Queue 5: Client(18, 15, 2);   
  
Time: 18  
Waiting clients: Client(19, 19, 1); Client(20, 19, 6); Client(21, 20, 4); Client(22, 21, 6); Client(23, 22, 5); Client(24, 22, 5); Client(25, 23, 7); Client(26, 23, 2); Client(27, 24, 7); Client(28, 24, 7); Client(29, 25, 1); Client(30, 26, 1); Client(31, 26, 6); Client(32, 28, 4); Client(33, 29, 6); Client(34, 29, 3); Client(35, 29, 4); Client(36, 29, 6); Client(37, 29, 4); Client(38, 30, 1); Client(39, 32, 4); Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: closed  
Queue 2: closed  
Queue 3: Client(17, 14, 1);   
Queue 4: closed  
Queue 5: Client(18, 15, 1);   
  
Time: 19  
Waiting clients: Client(21, 20, 4); Client(22, 21, 6); Client(23, 22, 5); Client(24, 22, 5); Client(25, 23, 7); Client(26, 23, 2); Client(27, 24, 7); Client(28, 24, 7); Client(29, 25, 1); Client(30, 26, 1); Client(31, 26, 6); Client(32, 28, 4); Client(33, 29, 6); Client(34, 29, 3); Client(35, 29, 4); Client(36, 29, 6); Client(37, 29, 4); Client(38, 30, 1); Client(39, 32, 4); Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: closed  
Queue 2: closed  
Queue 3: closed  
Queue 4: Client(19, 19, 1);   
Queue 5: Client(20, 19, 6);   
  
Time: 20  
Waiting clients: Client(22, 21, 6); Client(23, 22, 5); Client(24, 22, 5); Client(25, 23, 7); Client(26, 23, 2); Client(27, 24, 7); Client(28, 24, 7); Client(29, 25, 1); Client(30, 26, 1); Client(31, 26, 6); Client(32, 28, 4); Client(33, 29, 6); Client(34, 29, 3); Client(35, 29, 4); Client(36, 29, 6); Client(37, 29, 4); Client(38, 30, 1); Client(39, 32, 4); Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: Client(21, 20, 4);   
Queue 2: closed  
Queue 3: closed  
Queue 4: closed  
Queue 5: Client(20, 19, 5);   
  
Time: 21  
Waiting clients: Client(23, 22, 5); Client(24, 22, 5); Client(25, 23, 7); Client(26, 23, 2); Client(27, 24, 7); Client(28, 24, 7); Client(29, 25, 1); Client(30, 26, 1); Client(31, 26, 6); Client(32, 28, 4); Client(33, 29, 6); Client(34, 29, 3); Client(35, 29, 4); Client(36, 29, 6); Client(37, 29, 4); Client(38, 30, 1); Client(39, 32, 4); Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: Client(21, 20, 3);   
Queue 2: closed  
Queue 3: Client(22, 21, 6);   
Queue 4: closed  
Queue 5: Client(20, 19, 4);   
  
Time: 22  
Waiting clients: Client(25, 23, 7); Client(26, 23, 2); Client(27, 24, 7); Client(28, 24, 7); Client(29, 25, 1); Client(30, 26, 1); Client(31, 26, 6); Client(32, 28, 4); Client(33, 29, 6); Client(34, 29, 3); Client(35, 29, 4); Client(36, 29, 6); Client(37, 29, 4); Client(38, 30, 1); Client(39, 32, 4); Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: Client(21, 20, 2);   
Queue 2: Client(24, 22, 5);   
Queue 3: Client(22, 21, 5);   
Queue 4: Client(23, 22, 5);   
Queue 5: Client(20, 19, 3);   
  
Time: 23  
Waiting clients: Client(27, 24, 7); Client(28, 24, 7); Client(29, 25, 1); Client(30, 26, 1); Client(31, 26, 6); Client(32, 28, 4); Client(33, 29, 6); Client(34, 29, 3); Client(35, 29, 4); Client(36, 29, 6); Client(37, 29, 4); Client(38, 30, 1); Client(39, 32, 4); Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: Client(21, 20, 1); Client(25, 23, 7);   
Queue 2: Client(24, 22, 4);   
Queue 3: Client(22, 21, 4);   
Queue 4: Client(23, 22, 4); Client(26, 23, 2);   
Queue 5: Client(20, 19, 2);   
  
Time: 24  
Waiting clients: Client(29, 25, 1); Client(30, 26, 1); Client(31, 26, 6); Client(32, 28, 4); Client(33, 29, 6); Client(34, 29, 3); Client(35, 29, 4); Client(36, 29, 6); Client(37, 29, 4); Client(38, 30, 1); Client(39, 32, 4); Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: Client(25, 23, 7);   
Queue 2: Client(24, 22, 3); Client(28, 24, 7);   
Queue 3: Client(22, 21, 3);   
Queue 4: Client(23, 22, 3); Client(26, 23, 2);   
Queue 5: Client(20, 19, 1); Client(27, 24, 7);   
  
Time: 25  
Waiting clients: Client(30, 26, 1); Client(31, 26, 6); Client(32, 28, 4); Client(33, 29, 6); Client(34, 29, 3); Client(35, 29, 4); Client(36, 29, 6); Client(37, 29, 4); Client(38, 30, 1); Client(39, 32, 4); Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: Client(25, 23, 6);   
Queue 2: Client(24, 22, 2); Client(28, 24, 7);   
Queue 3: Client(22, 21, 2); Client(29, 25, 1);   
Queue 4: Client(23, 22, 2); Client(26, 23, 2);   
Queue 5: Client(27, 24, 7);   
  
Time: 26  
Waiting clients: Client(32, 28, 4); Client(33, 29, 6); Client(34, 29, 3); Client(35, 29, 4); Client(36, 29, 6); Client(37, 29, 4); Client(38, 30, 1); Client(39, 32, 4); Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: Client(25, 23, 5);   
Queue 2: Client(24, 22, 1); Client(28, 24, 7);   
Queue 3: Client(22, 21, 1); Client(29, 25, 1); Client(30, 26, 1);   
Queue 4: Client(23, 22, 1); Client(26, 23, 2); Client(31, 26, 6);   
Queue 5: Client(27, 24, 6);   
  
Time: 27  
Waiting clients: Client(32, 28, 4); Client(33, 29, 6); Client(34, 29, 3); Client(35, 29, 4); Client(36, 29, 6); Client(37, 29, 4); Client(38, 30, 1); Client(39, 32, 4); Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: Client(25, 23, 4);   
Queue 2: Client(28, 24, 7);   
Queue 3: Client(29, 25, 1); Client(30, 26, 1);   
Queue 4: Client(26, 23, 2); Client(31, 26, 6);   
Queue 5: Client(27, 24, 5);   
  
Time: 28  
Waiting clients: Client(33, 29, 6); Client(34, 29, 3); Client(35, 29, 4); Client(36, 29, 6); Client(37, 29, 4); Client(38, 30, 1); Client(39, 32, 4); Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: Client(25, 23, 3);   
Queue 2: Client(28, 24, 6);   
Queue 3: Client(30, 26, 1); Client(32, 28, 4);   
Queue 4: Client(26, 23, 1); Client(31, 26, 6);   
Queue 5: Client(27, 24, 4);   
  
Time: 29  
Waiting clients: Client(38, 30, 1); Client(39, 32, 4); Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: Client(25, 23, 2); Client(33, 29, 6);   
Queue 2: Client(28, 24, 5); Client(36, 29, 6);   
Queue 3: Client(32, 28, 4); Client(34, 29, 3);   
Queue 4: Client(31, 26, 6); Client(37, 29, 4);   
Queue 5: Client(27, 24, 3); Client(35, 29, 4);   
  
Time: 30  
Waiting clients: Client(39, 32, 4); Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: Client(25, 23, 1); Client(33, 29, 6);   
Queue 2: Client(28, 24, 4); Client(36, 29, 6);   
Queue 3: Client(32, 28, 3); Client(34, 29, 3); Client(38, 30, 1);   
Queue 4: Client(31, 26, 5); Client(37, 29, 4);   
Queue 5: Client(27, 24, 2); Client(35, 29, 4);   
  
Time: 31  
Waiting clients: Client(39, 32, 4); Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: Client(33, 29, 6);   
Queue 2: Client(28, 24, 3); Client(36, 29, 6);   
Queue 3: Client(32, 28, 2); Client(34, 29, 3); Client(38, 30, 1);   
Queue 4: Client(31, 26, 4); Client(37, 29, 4);   
Queue 5: Client(27, 24, 1); Client(35, 29, 4);   
  
Time: 32  
Waiting clients: Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: Client(33, 29, 5);   
Queue 2: Client(28, 24, 2); Client(36, 29, 6);   
Queue 3: Client(32, 28, 1); Client(34, 29, 3); Client(38, 30, 1); Client(39, 32, 4);   
Queue 4: Client(31, 26, 3); Client(37, 29, 4);   
Queue 5: Client(35, 29, 4);   
  
Time: 33  
Waiting clients: Client(40, 34, 4); Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: Client(33, 29, 4);   
Queue 2: Client(28, 24, 1); Client(36, 29, 6);   
Queue 3: Client(34, 29, 3); Client(38, 30, 1); Client(39, 32, 4);   
Queue 4: Client(31, 26, 2); Client(37, 29, 4);   
Queue 5: Client(35, 29, 3);   
  
Time: 34  
Waiting clients: Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: Client(33, 29, 3);   
Queue 2: Client(36, 29, 6);   
Queue 3: Client(34, 29, 2); Client(38, 30, 1); Client(39, 32, 4);   
Queue 4: Client(31, 26, 1); Client(37, 29, 4);   
Queue 5: Client(35, 29, 2); Client(40, 34, 4);   
  
Time: 35  
Waiting clients: Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: Client(33, 29, 2);   
Queue 2: Client(36, 29, 5);   
Queue 3: Client(34, 29, 1); Client(38, 30, 1); Client(39, 32, 4);   
Queue 4: Client(37, 29, 4);   
Queue 5: Client(35, 29, 1); Client(40, 34, 4);   
  
Time: 36  
Waiting clients: Client(41, 37, 5); Client(42, 37, 3); Client(43, 37, 6); Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: Client(33, 29, 1);   
Queue 2: Client(36, 29, 4);   
Queue 3: Client(38, 30, 1); Client(39, 32, 4);   
Queue 4: Client(37, 29, 3);   
Queue 5: Client(40, 34, 4);   
  
Time: 37  
Waiting clients: Client(44, 38, 7); Client(45, 38, 5); Client(46, 38, 5); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: Client(41, 37, 5);   
Queue 2: Client(36, 29, 3); Client(43, 37, 6);   
Queue 3: Client(39, 32, 4);   
Queue 4: Client(37, 29, 2); Client(42, 37, 3);   
Queue 5: Client(40, 34, 3);   
  
Time: 38  
Waiting clients: Client(48, 39, 1); Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: Client(41, 37, 4); Client(47, 38, 1);   
Queue 2: Client(36, 29, 2); Client(43, 37, 6);   
Queue 3: Client(39, 32, 3); Client(44, 38, 7);   
Queue 4: Client(37, 29, 1); Client(42, 37, 3); Client(45, 38, 5);   
Queue 5: Client(40, 34, 2); Client(46, 38, 5);   
  
Time: 39  
Waiting clients: Client(49, 40, 7); Client(50, 40, 6);   
Queue 1: Client(41, 37, 3); Client(47, 38, 1); Client(48, 39, 1);   
Queue 2: Client(36, 29, 1); Client(43, 37, 6);   
Queue 3: Client(39, 32, 2); Client(44, 38, 7);   
Queue 4: Client(42, 37, 3); Client(45, 38, 5);   
Queue 5: Client(40, 34, 1); Client(46, 38, 5);   
  
Time: 40  
Waiting clients:   
Queue 1: Client(41, 37, 2); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7);   
Queue 2: Client(43, 37, 6); Client(50, 40, 6);   
Queue 3: Client(39, 32, 1); Client(44, 38, 7);   
Queue 4: Client(42, 37, 2); Client(45, 38, 5);   
Queue 5: Client(46, 38, 5);   
  
Time: 41  
Waiting clients:   
Queue 1: Client(41, 37, 1); Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7);   
Queue 2: Client(43, 37, 5); Client(50, 40, 6);   
Queue 3: Client(44, 38, 7);   
Queue 4: Client(42, 37, 1); Client(45, 38, 5);   
Queue 5: Client(46, 38, 4);   
  
Time: 42  
Waiting clients:   
Queue 1: Client(47, 38, 1); Client(48, 39, 1); Client(49, 40, 7);   
Queue 2: Client(43, 37, 4); Client(50, 40, 6);   
Queue 3: Client(44, 38, 6);   
Queue 4: Client(45, 38, 5);   
Queue 5: Client(46, 38, 3);   
  
Time: 43  
Waiting clients:   
Queue 1: Client(48, 39, 1); Client(49, 40, 7);   
Queue 2: Client(43, 37, 3); Client(50, 40, 6);   
Queue 3: Client(44, 38, 5);   
Queue 4: Client(45, 38, 4);   
Queue 5: Client(46, 38, 2);   
  
Time: 44  
Waiting clients:   
Queue 1: Client(49, 40, 7);   
Queue 2: Client(43, 37, 2); Client(50, 40, 6);   
Queue 3: Client(44, 38, 4);   
Queue 4: Client(45, 38, 3);   
Queue 5: Client(46, 38, 1);   
  
Time: 45  
Waiting clients:   
Queue 1: Client(49, 40, 6);   
Queue 2: Client(43, 37, 1); Client(50, 40, 6);   
Queue 3: Client(44, 38, 3);   
Queue 4: Client(45, 38, 2);   
Queue 5: closed  
  
Time: 46  
Waiting clients:   
Queue 1: Client(49, 40, 5);   
Queue 2: Client(50, 40, 6);   
Queue 3: Client(44, 38, 2);   
Queue 4: Client(45, 38, 1);   
Queue 5: closed  
  
Time: 47  
Waiting clients:   
Queue 1: Client(49, 40, 4);   
Queue 2: Client(50, 40, 5);   
Queue 3: Client(44, 38, 1);   
Queue 4: closed  
Queue 5: closed  
  
Time: 48  
Waiting clients:   
Queue 1: Client(49, 40, 3);   
Queue 2: Client(50, 40, 4);   
Queue 3: closed  
Queue 4: closed  
Queue 5: closed  
  
Time: 49  
Waiting clients:   
Queue 1: Client(49, 40, 2);   
Queue 2: Client(50, 40, 3);   
Queue 3: closed  
Queue 4: closed  
Queue 5: closed  
  
Time: 50  
Waiting clients:   
Queue 1: Client(49, 40, 1);   
Queue 2: Client(50, 40, 2);   
Queue 3: closed  
Queue 4: closed  
Queue 5: closed  
  
Time: 51  
Waiting clients:   
Queue 1: closed  
Queue 2: Client(50, 40, 1);   
Queue 3: closed  
Queue 4: closed  
Queue 5: closed  
  
Time: 52  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
Queue 3: closed  
Queue 4: closed  
Queue 5: closed  
  
Time: 53  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
Queue 3: closed  
Queue 4: closed  
Queue 5: closed  
  
Time: 54  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
Queue 3: closed  
Queue 4: closed  
Queue 5: closed  
  
Time: 55  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
Queue 3: closed  
Queue 4: closed  
Queue 5: closed  
  
Time: 56  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
Queue 3: closed  
Queue 4: closed  
Queue 5: closed  
  
Time: 57  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
Queue 3: closed  
Queue 4: closed  
Queue 5: closed  
  
Time: 58  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
Queue 3: closed  
Queue 4: closed  
Queue 5: closed  
  
Time: 59  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
Queue 3: closed  
Queue 4: closed  
Queue 5: closed  
  
Time: 60  
Waiting clients:   
Queue 1: closed  
Queue 2: closed  
Queue 3: closed  
Queue 4: closed  
Queue 5: closed  
  
Average Waiting Time: 1.74  
Average Service Time: 4.08  
Peak Hour: 30

CONCLUSIONS

This assignment helped me understand a little more better how to work with multiple threads and how to make sure we protect critical zones (variables was mostly in our case) such that we don’t lose or modify essetial data.

BIBLIOGRAPHY

<https://mkyong.com/java/java-generate-random-integers-in-a-range/>

<https://stackoverflow.com/questions/18448671/how-to-avoid-concurrentmodificationexception-while-removing-elements-from-arr>