Project p2_40354020_172

The purpose of this project is to analyze the differences in waiting line structures similar to those you would find in a fast food restaurant, a supermarket, a toll, etc. We will study four ways to handle waiting lines and compare their average waiting times per customer, the times to finish and the amount of people that were attended before a customer who arrived earlier. The four strategies are as follows:

1. SLMS: There is only one waiting line and multiple servers. Whenever a server is available the next customer in line will start being served.
2. MLMS: There are as many lines as there are servers. Each line corresponds to a server and may not transfer to a different line of be attended by a different server. When new customers arrive, they will be placed in the shortest line.
3. MLMSBLL: Works exactly as the previous strategy, except that the people at the end of each line may be transferred to the shortest line that they may benefit from (if such line exists). If more than one customer benefits from a line transfer, they will be transferred by order of who was first in the input file.
4. MLMSBWT: Works as MLMS except that the new arrivals will not be placed at the shortest line. They will be placed in the line with the lowest expected waiting time. No line transfer is allowed.

In the project directory there will be a folder containing many files, each representing a set of customers with arrival times and service times (time it takes for the customer to be served). They will be in increasing order by arrival time. The program will take each file and run a simulation in which the customers will be tested using al four strategies, and for each strategy using 1, 3 and 5 servers. Then it will create an output file for each file with the results of the specified file. If the file was not found or had an incorrect format, the output file will contain the error. It is essential that the file containing the names of all the other files is in the folder so that the program knows which files to look for.

Instructions to execute Main from the terminal windows

To execute the program from the terminal, specifically the Main class which runs the program, you must first go to the project directory. Make sure that the file “dataFiles.txt” is in the inputFiles folder within the directory, along with the “data_i.txt” files that are going to serve as the input. Once in the project directory, execute the following command:

java -classpath ./bin serviceClasses/Main

Now, in the inputFiles folder there should be an output file data_i_OUT.txt for each file data_i.txt.