

# CS 150: Project II

## Project Description

A popular coffee shop chain is opening a new coffee shop on College Hill. You are hired as a consultant to decide how many cashiers to hire in order to optimize the profit. The following conditions are known.

1. The shop opens at 6 am and closed at 9 pm.
2. There are  $s$  cashier counters staffed in the shop.
3. When a customer arrives at the shop, he or she joins a single queue waiting to be served by the first available cashier. However, if the length of the queue is  $8 \times s$  or more, the customer is turned away as an “overflow.” When the shop closes at 9 pm, no new customer can enter the shop but the customers already in the queue will still be served.
4. The estimated profit of serving each customer is a random number between  $p_1$  and  $p_2$  dollars, inclusive.
5. The cost of staffing a cashier counter for a whole day is  $c$  dollars.
6. The estimated time required to serve a customer is a random number between  $t_1$  and  $t_2$  seconds, inclusive.
7. The daily net profit of the shop is the total daily profit minus the daily cost of cashiers.

## Input Specification

You are given a file “input.txt” containing the following information.

- The first line are two real numbers  $p_1$  and  $p_2$ ,  $0.00 < p_1 \leq p_2 < 10.00$ , the lower and upper bounds of the estimated profit of serving each customer in dollars.
- The second line is a real number  $c$ ,  $0.00 < c < 1000.00$ , the cost of staffing a cashier counter per day in dollars.
- The third line are two integers  $t_1$  and  $t_2$ ,  $30 < t_1 \leq t_2 < 300$ , the lower and upper bounds of the average time for a cashier to serve a customer in seconds.
- From the fourth line to the last line is the arrival times of the customers. Each line contains a string in the format of “hh:mm:ss” in the 24-hour format.

## Assignments

As a consultant, your assignments are:

1. Write a program that can compute, for the give data file and an integer parameter  $s$  ( $s > 0$  is the number of cashier counters staffed) read from the standard input, the amount of daily net profit, the rate of “overflow”, the average and maximum waiting time of all customers served.

2. Run the program multiple times to compute the average net daily profit for different values of the number of cashiers  $s$ .
3. Plot the average net daily profit as a function of  $s$  and find the optimal number of cashiers  $s$  that maximizes the profit.

## Guidelines

1. You should use the appropriate data structures and algorithms. Specifically, the following data structures are used in this project: ArrayList, LinkedList (or ArrayDeque), Queue, PriorityQueue.
2. You will use a FIFO queue to store the customers who are waiting to be served.
3. You will use a priority queue to store the events that are waiting to be processed in order of time, including customer arrivals and departures. This is called an “event-driven” simulation. Initially, the customer arrival events are stored in a priority queue in order of time. When a customer is served, the customer’s departure time is computed, and the departure event is inserted into the priority queue. In each iteration of the simulation, the event on top of the priority queue is removed and processed. **Read the textbook section 13.32 to understand event-driven simulations before attempting the project.**
4. Your conclusions and analysis should be supported by data from your simulation.

## Report

The guideline for wiring the project report is on Moodle

## Grading

The project is to be completed individually. Unless otherwise approved by the instructor, the only person you can consult is the instructor. Your project will be graded on the following criteria (assuming the program compiles and runs):

1. correctness of the program (your program will be tested on a specific test input file to check for correctness)
2. documentation (methods and classes) including Javadoc
3. unit testing
4. object oriented design
5. quality of the simulation and analysis
6. quality of the project report