

GNG1106 Lab 4

Simulation of a Fast-Food Restaurant

Designed By Yongyi Mao

Description

In this lab, you are to simulate the service of customers in a fast-food restaurant, and investigate the behaviour therein.

The restaurant

Suppose that a fast-food restaurant sells salad and burger. There are two cashiers. With cashier 1, the number of seconds that it takes to complete an order of salad is uniformly distributed in $\{55, 56, \dots, 64, 65\}$; and the number of seconds it takes to complete an order of burger is uniformly distributed in $\{111, 112, \dots, 129, 130\}$. With cashier 2, the number of seconds that it takes to complete an order of salad is uniformly distributed in $\{65, 66, \dots, 74, 75\}$; and the number of seconds it takes to complete an order of burger is uniformly distributed in $\{121, 122, \dots, 139, 140\}$.

Assume that the customers arrive at random times but has an average arrival rate of r customers per minute. Consider two different scenarios.

- Customers wait in one line for service, and when either of two cashiers is available, the first customer in the line goes to the cashier and gets serviced. In this scenario, when a customer arrives at the restaurant, he either gets serviced if there is no line up, or wait at the end of the line.
- Customers wait in two lines, each for a cashier. The first customer in a line will get serviced if and only if the cashier for his line becomes available. In this scenario, when a customer arrives at the restaurant, he joins the shorter line. In addition, we impose the condition that if a customer joins a line, he will not move to the other line or to the other cashier when the other line becomes shorter or when the other cashier becomes free.

In both scenarios considered, a cashier will only start serving the next customer when the customer he is currently serving has received his ordered food. (That is the point we call “the customer’s order is completed”.)

Simulation

For each of the two scenarios and for several choices of r (see later description), you are to simulate the customers arriving/waiting/getting service over a period of 3 hours, namely, from time 0 to time 180 minutes, where you assume that at time 0 there is no customer waiting and both cashiers are available;

The entire period of 3 hours is to be divided into time slots each of 1 second duration. At each time slot, with $r/60$ probability, you make one new customer arrive, and with $1 - r/60$ probability you make no new customer arrive. This should give rise to an average customer arrival rate of r customers/minute, and the arrival model will be reasonably close to what is described above. In each time slot, you will make your program handle whatever necessary, in order to achieve the objectives of the lab.

Objectives and deliverables

You need to write two programs to investigate the following.

1. For each of the two scenarios and for each r (see 2 below), you are to divide the three-hour simulated period into 10-minute periods, and for every customer arriving during period i ($i \in \{1, 2, \dots, 18\}$), compute the overall waiting time of the customer (namely, from the time he arrives at the restaurant to the time when his order is completed). You need to print for each i the average waiting time for the customers arriving during period i . Note that if a customer arriving in period i has not been served within the three-hour simulated period, then his waiting time is not known. So the average waiting time for customers arriving in this period can not be computed. In that case, simply print “not available” as the average waiting time for that period.
2. For each scenario, pick a few values of r and plot the average waiting time as a function of i (for those i for which the waiting time is computable) on graph papers. Use the values of r that give rise to different trends of the graphed function.
3. Note that for each scenario, there should be a threshold value of r beyond which the average customer waiting time becomes longer and longer. This corresponds to the case where the arrival rate of the customers exceeds the service capacity of the restaurant. Let your program estimate and print, for each scenario, the threshold value of r .

Your first program is expected to answer the above questions 1 and 2, and your second program is expected to answer question 3, `bf` programatically.

Deliverables include

- the code of your programs
- the running results of your programs (to be inspected by the TA)
- graphs (hand-made).

Organization of the Code

As before, common functions need to be extracted out and put into a separate file. Organize your code as cleanly as possible.

Pre-Lab Preparation

Make a flow chart for each of your programs, and list all the function prototypes that you will create for this lab. Show your work to the TA.

Grading Notes

- In addition to writing code with correct logic and syntax, your programming style will also be evaluated. A good programming style includes, for now, necessary indentation, sufficient comments, good naming convention, ease of reading, clean structure and well organized code into functions.
- Overall grading scheme (out of 100 points):
 - Pre-Lab preparation: 10 points
 - Programming Style: 20 points
 - Syntax: 30 points

- Logic, and running, and everything else: 40 points
- Bonus points (up to 10) may be given to particularly smart and elegant code