Assignment 1

Server Queue Simulation

Description

In this assignment you will implement simulation models for M/M/K server queue system. In the previous class, step by step implementation guideline for M/M/1 system has been demonstrated with a program written in “C”. For this assignment you will be provided a python skeleton code (mmk.py). <u>It is highly recommended that you use this python skeleton for this assignment.</u> If you want to use the “C” code you should first convert it in “C++” project. In the python skeleton, codes for potting graphs (you have to show some graphs for this assignment) have already been provided. If you use “C++”, you have to use graph plotting libraries (like https://github.com/lava/matplotlib-cpp).

Terminology

Lambda (**λ**) = Interarrival rate

Mu (**μ**) = Service rate

Ro (**ρ**) = **λ / μ**

Average Queue Length (**L**)

Average Delay in Queue (**D**)

Server Utilization Factor (**U**)

Number of servers (**k**)

Note that **λ**and **μ** are rates, not mean value.

Experiment 1

This is basic M/M/1 implementation. You have to compare your experimental results with analytic results for this experiment.

Analytic results

Average Queue Length (L) = $\frac{\lambda^2}{\mu(\mu-\lambda)}$

Average Delay in Queue (D) = $\frac{\lambda}{\mu(\mu-\lambda)}$

Server Utilization Factor(U)= $\frac{\lambda}{\mu}$

Experiment 2

It is also a variant of M/M/1 system. You will run experiment 1 for different values of ρ (see mmk.py). You will have to draw the following figures using experimental results (see mmk.py).

1. L vs ρ
2. D vs ρ
3. U vs ρ

Experiment 3

In this experiment ρ is fixed and number of servers (k) will be varied (see mmk.py). We can call this M/M/K system. Like experiment 2 you will have to draw the following figures using experimental results.

1. L vs k
2. D vs k
3. U vs k

Experiment 4

In the previous experiments there was a single queue. In this experiment number of queues will be equal to the number of servers which means each server will have its own queue. There will be some changes in the arrival and departure subroutine.

Arrival

Upon arrival if a customer finds all the servers busy, he will move in the leftmost shortest queue. Everything else is same as single queue system.

Departure

When a customer leaves a server, the next person standing in corresponding server's queue will be served. Now assume after departure length of this server's queue is ***L***. Also assume length of the left and right queues are ***LF*** and ***LR***respectively.If either ***(LF – L)*** or ***(LR – L)*****>= 2** then one or more customers from the tail of the longer queue will join the current server's queue. If one of them finds the server idle, he/she will be served immediately.

In this experiment vary the values of number of servers (k) like experiment 3 and draw the following diagrams using experimental results.

1. L vs k
2. D vs k
3. U vs k

For finding average queue length, calculate average queue length of each queue and calculate their mean value.

Submission Guideline

1. For each experiment create a py or cpp file named experiment_k.py or experiment_k.cpp where k = 1,2,3 or 4
2. Create a folder with your 7-digit student ID (1505XXX) and put all these files and additional files (if any) in that folder. Zip this folder and name of the zip file will be (1505XXX.zip). Finally submit the zip fie in Moodle.
3. Deadline 21/03/2020 (11:59 PM)
4. Plagiarism is strictly prohibited. You will get -100% if you are caught.