CAPACITY ANALYSIS

PROJECT REPORT

BY

Badam Alivelu Laxmi Mani Harika

Student no.: 20000628-T083

INDEX

* Introduction
* Scenario
* Aim
* Materials used
* Analysis
* Results
* Appendix
* References

INTRODUCTION

Until now we have seen many cases where the customers eventually get served. But in our daily life, we come across some impatient customers who initially join the queue, but leave the queue without being served if it is taking a longer time (some finite value). In queueing theory, there is a term reneging which means joining the queue but leaving without taking the service. This type of customers is a problem in business field as it can lead to a huge loss in the revenue of the business.

Here, we are going to see one such case of M/M/1 system where all the customers are of this type.

SCENARIO

At a busy metro station, there is only one metro train ticket counter. Assume, arrivals to this counter are following a poisson distribution with a rate of 25 customers per hour (arrivals are independent of time) and service times are following an exponential distribution with rate of 40 customers per hour.

This ticket counter also has a fixed queue length of 6 customers and works for 10 hours a day without any break.

Customers will wait in the queue for at most a threshold value (Tv), which follows a uniform distribution between 8 minutes to 10 minutes. If they did not receive the service within this time, they simply leave the queue without getting service.

The manager has noticed that people are leaving the system after joining the queue and realizes that this is directly affecting the revenue of their business and wants to know how many people are leaving this way and what is the fraction of people who are leaving this way, in one day.

After finding that, he wants to find out what is the minimum service rate he has to maintain if the fraction of customers who are leaving the queue without taking service should not exceed 0.1 if the maximum service rate, he can provide is 0.84.

AIM

To observe the one-day activity at the metro ticket counter and finding out:

1) The number of customers who are leaving the queue without taking the service. And also finding out what fraction of the customers are leaving the queue this way in one day, using MATLAB.

2) Minimum service rate to be maintained so that the fraction of customers who leave the queue without getting service should not exceed 0.1.

MATERIALS USED

* Matlab tools, functions and documentation
* Problem set-8 (Qps8.m)

ANALYSIS

Information given:

The system given is a M/M/1/7 and the discipline followed here is FCFS (First Come First Serve).

Arrival rate = 25/60 = 0.4167

Departure rate = 40/60 = 0.6667

End time (in minutes) = 600

Maximum queue length = 6

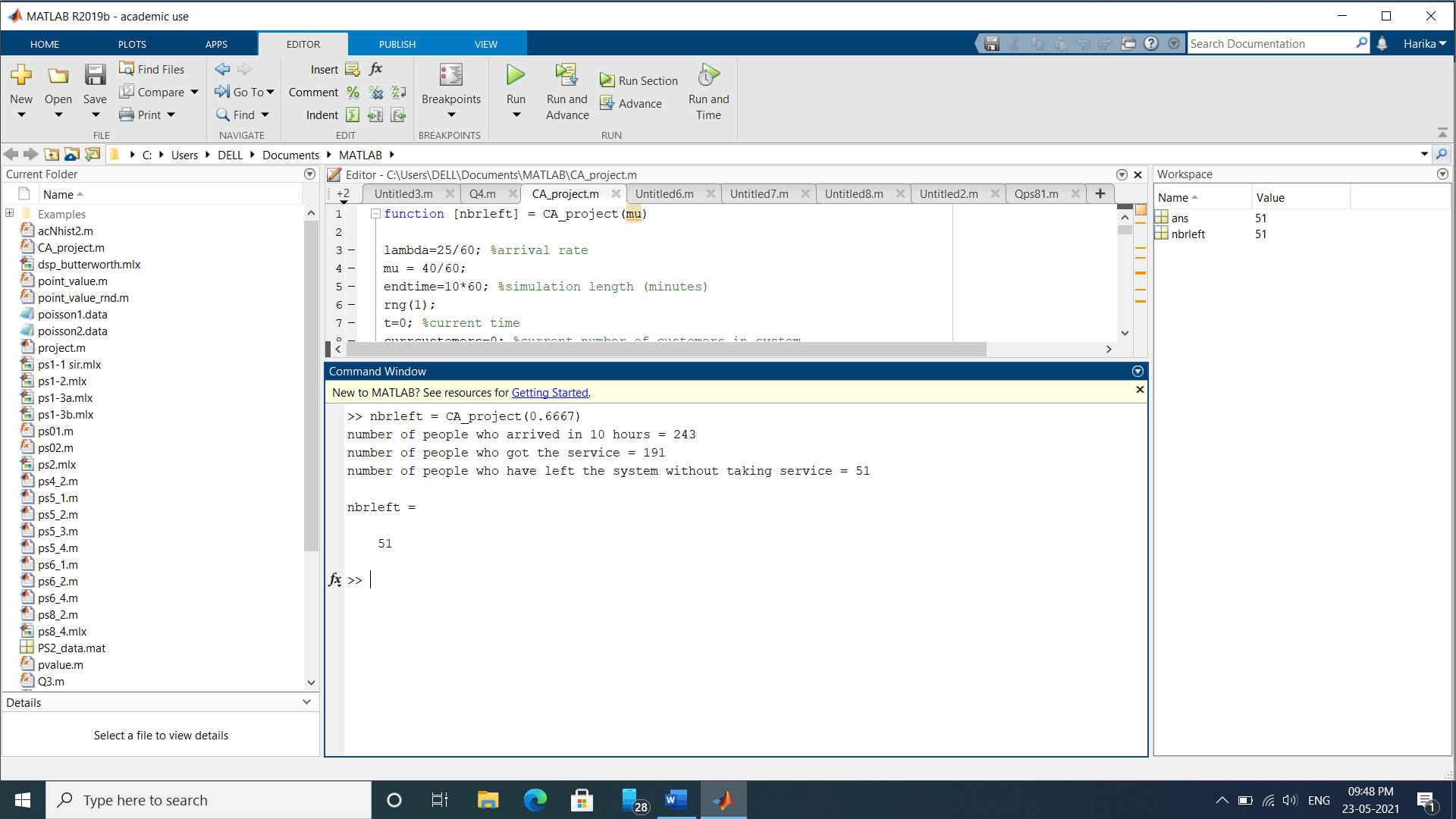
Method:

At regular time intervals (which is a uniformly distributed random value between 1 and 2), I am calculating the waiting time of all the customers in the queue. If any of the customers waiting time is greater than the threshold value (which is a uniformly distributed random value between 8 to 10 minutes), then the departed time of that customer is updated.

For calculating the minimum service time so that the number of customers who left the system without taking service to be less than 5, I have plotted a graph for all the mu values (between 0.6667 and 0.84) and number of customers who left. Using these values, I have calculated the minimum mu value.

RESULTS

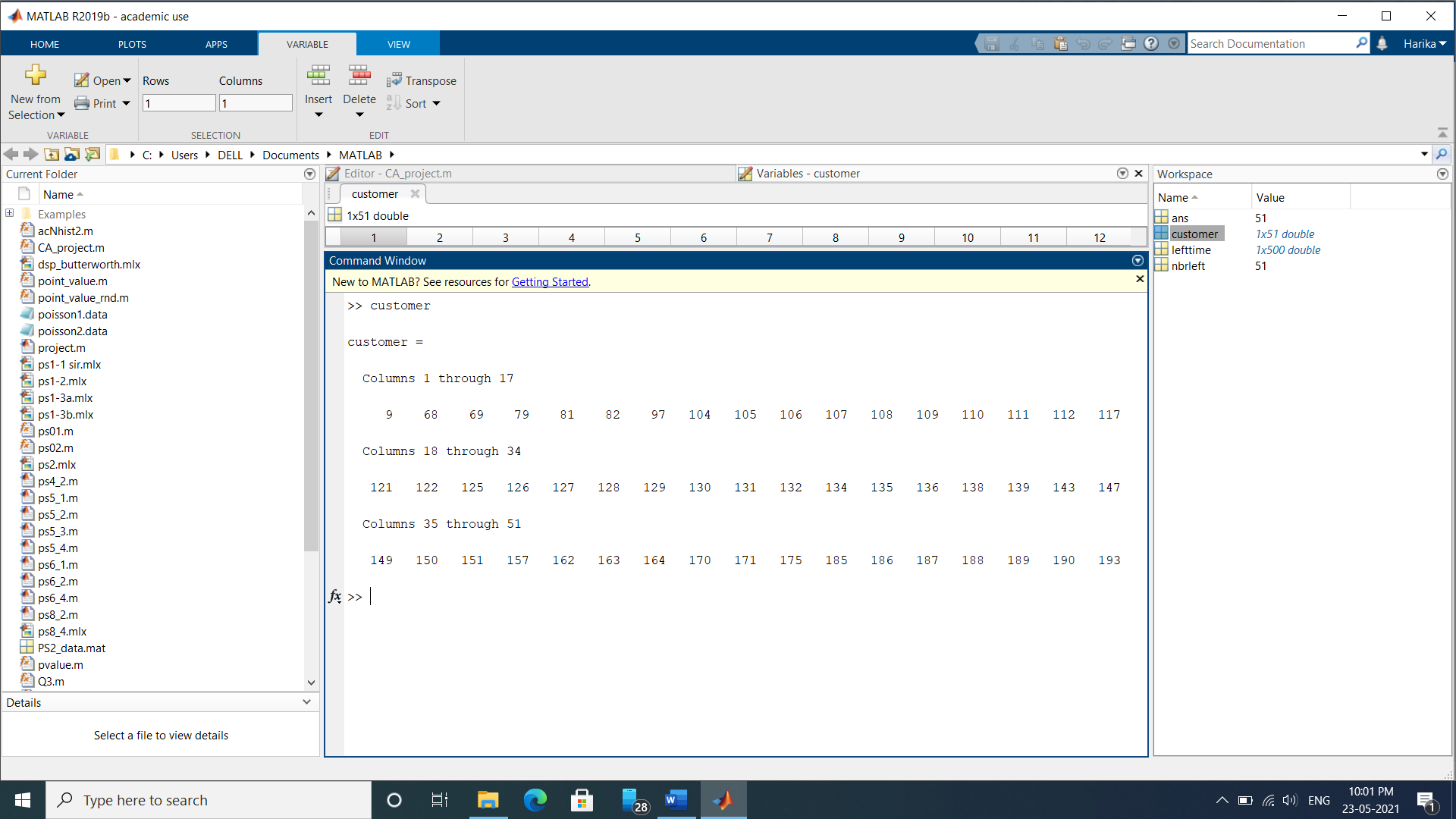
1)



Number of customers who left without taking the service = 51

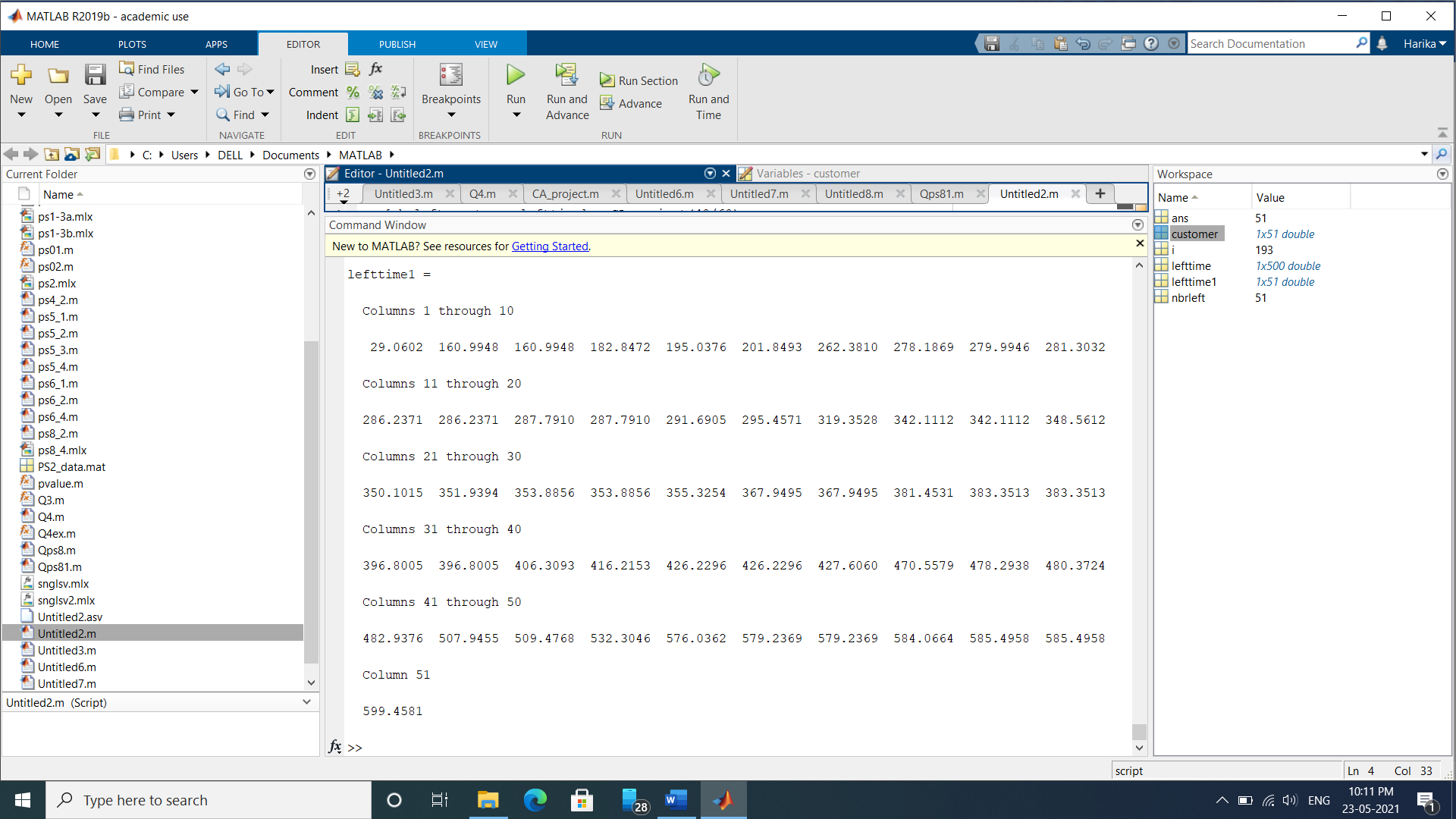
Fraction of customers who left the queue without taking the service = 51/243

= 0.2099



Above values shows the customers who have left.

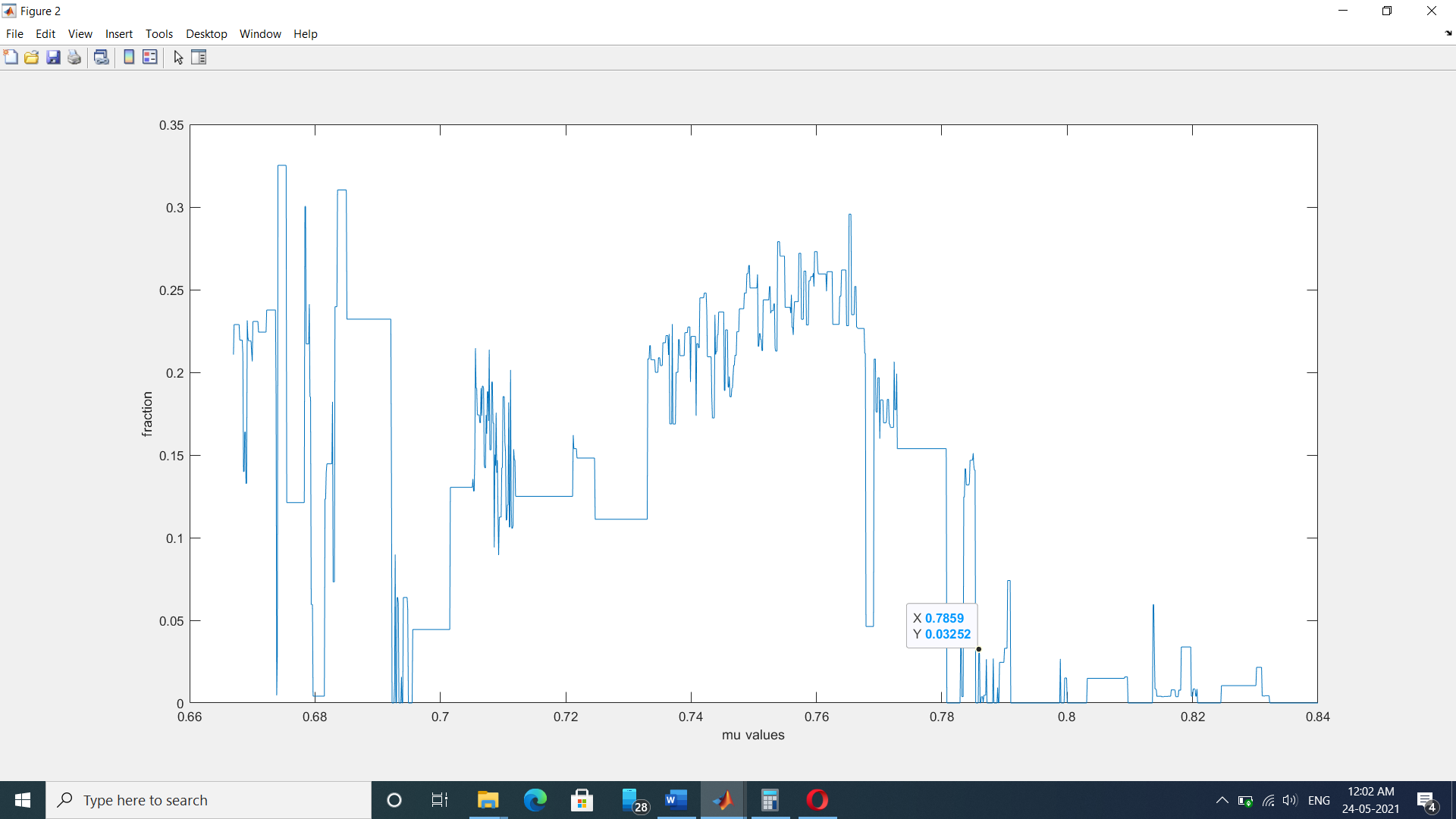
(for example: 9th customer has left the queue without taking the service)



Above values shows the time at which the customers left without taking the service.

(for example: 9th customer left the queue without taking service at 29.0602 minutes)

2)



The above graph is plotted between mu values and fraction values.

Minimum mu value to be maintained so that the fraction of customers who leave the queue without getting served should not exceed 0.1 is 0.7859.

APPENDIX

Function used:

function [nbrleft,nbrdeparted,customer,lefttime] = function1(mu)

lambda=25/60; %arrival rate

endtime=10\*60; %simulation length (minutes)

rng(1);

t=0; %current time

currcustomers=0; %current number of customers in system

event=zeros(1,3); %constructs vector to keep time

event(1)=exprnd(1/lambda); %time for next arrival1

event(2)=inf; %no next service completion (empty system)

event(3)=1+rand; %time for calculating the waiting time of customers in the queue

nbrdeparted=0; %number of departed customers

nbrarrived=0; %number of customers that have arrived throughout the simulations

maxtime = 8+2\*rand; %maximum time a customers waits in the queue - uniformly distributed between 8 to 10 minutes

nbrleft = 0; %number of people who left the queue without taking service

lefttime = zeros(1,500); %time at which people left the queue without taking service

queuelength = 6; %maximum queue length

customer = []; %tells which arrival is leaving the queue without taking service

departedtime = [];

while t<endtime

[t,nextevent]=min(event);

if nextevent==1 && currcustomers <= queuelength+1

event(1)=exprnd(1/lambda)+t;

currcustomers=currcustomers+1;

nbrarrived=nbrarrived+1; %one more customer has arrived to the system through the simulations

arrivedtime(nbrarrived)=t; %the new customer arrived at time t

if currcustomers==1

event(2)=exprnd(1/mu)+t;

end

if currcustomers == queuelength +1

event(1) = inf;

end

elseif nextevent==2

currcustomers=currcustomers-1;

timeinsystem=t-arrivedtime(nbrarrived-currcustomers);

nbrdeparted=nbrdeparted+1;

departuretime(nbrdeparted) = t;%one more customer has departed%from the system through the%simulations

T(nbrdeparted)=timeinsystem; %timeinsystem of the customers who took the service

if currcustomers>0

event(2)=exprnd(1/mu)+t;

else

event(2)=inf;

end

if currcustomers == queuelength

event(1) = t + exprnd(1/lambda);

end

else

if currcustomers > 1

for i = 1: nbrarrived

waitingtime(i) = t - arrivedtime(i); %waiting time of all the customers who have arrived till now

end

waitingtime(1:nbrdeparted+1) = 0; %waiting time of the customers who have departed already

for i = nbrdeparted+2 : length(waitingtime)

if waitingtime(i) >= maxtime && currcustomers > 1 && lefttime(i) == 0

nbrleft = nbrleft + 1;

lefttime(i) = t;

currcustomers = currcustomers - 1;

customer = [customer,i];

i = i+1;

end

end

end

event(3) = t + rand + 1;

maxtime = 2\*rand + 8;

end

end

fprintf('number of people who arrived in 10 hours = %d', nbrarrived);

fprintf('\nnumber of people who got the service = %d', nbrdeparted);

fprintf('\nnumber of people who have left the system without taking service = %d\n', nbrleft);

end

1)

[nbrleft,nbrdeparted,customer,lefttime] = function1(40/60)

2)

x = 0.667 :0.0001: 0.84;

for i = 1: length(x)

[nbrleft(i), nbrdeparted(i)] = function1(x(i));

end

fraction = (nbrleft./(nbrleft + nbrdeparted));

figure;plot(x,fraction);

xlabel('mu values');

ylabel('fraction');

REFERENCES

* Lecture PPTs and problem sets provided by Siamak Khatibi
* https://www.researchgate.net/publication/307742156\_MM1N\_Queuing\_System\_with\_Retention\_of\_Reneged\_Customers