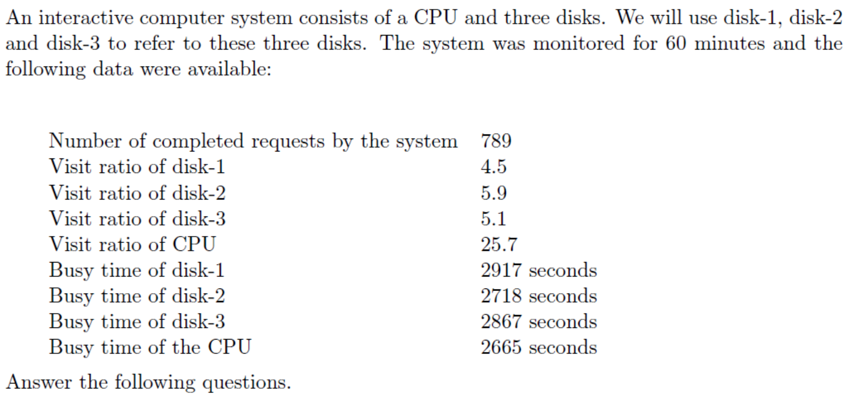
COMP9334 - Capacity Planning of Computer Systems and Networks

T1 2022

Assignment 1

Name: Yuhua Zhao – **ZID:** z5404443

Question 1 (3 Marks):



An interactive computer system consists of a CPU and three disks. We will use disk-1, disk-2 and disk-3 to refer to these three disks. The system was monitored for 60 minutes and the following data were available:

Number of completed requests by the system	789
Visit ratio of disk-1	4.5
Visit ratio of disk-2	5.9
Visit ratio of disk-3	5.1
Visit ratio of CPU	25.7
Busy time of disk-1	2917 seconds
Busy time of disk-2	2718 seconds
Busy time of disk-3	2867 seconds
Busy time of the CPU	2665 seconds

Answer the following questions.

a) *Determine the service demands of disk-1, disk-2, disk-3 and the CPU.*

To get the Service demand of Disk-1, Disk-2, disk-3, and the CPU, we need to use the Service Demand Law:

$$\boldsymbol{D(j) = \frac{U(i)}{X(0)}}$$

The U(i) refer to the Utilization of devices, which mean that we need to calculate the Utilization of U(Disk-1), U(Disk-2), U(Disk-3) and U(CPU) and the Throughput of the System. Meanwhile we will convert the Monitor time from minute to second to match with the provided Busy time.

$$U(Disk1) = \frac{B(Disk1)}{T} = \frac{2917}{60*60} \approx 0.810 \qquad U(Disk2) = \frac{B(Disk2)}{T} = \frac{2718}{60*60} = 0.755$$

$$U(Disk3) = \frac{B(Disk3)}{T} = \frac{2867}{60*60} \approx 0.796 \qquad U(CPU) = \frac{B(CPU)}{T} = \frac{2665}{60*60} \approx 0.740$$

After calculate the Utilization of each devices, we need to get the Throughput of the System X(0).

$$X(0) = \frac{C(0)}{T} = \frac{789}{60*60} \approx\ 0.219\ (Complection/S)$$

Since we retrieve the X(0) and the Utilization of each devices, so we can calculate the Service Demand of each device of the System.

$$D(Disk1) = \frac{U(Disk1)}{X(0)} = \frac{0.810}{0.219} = 3.698 \qquad D(Disk2) = \frac{U(Disk2)}{X(0)} = \frac{0.755}{0.219} = 3.447$$

$$D(Disk3) = \frac{U(Disk3)}{X(0)} = \frac{0.796}{0.219} = 3.635 \qquad D(CPU) = \frac{U(CPU)}{X(0)} = \frac{0.740}{0.219} = 3.379$$

b) *Use bottleneck analysis to determine the asymptotic bound on the system throughput when there are 4 interactive users, and the think time is 20 seconds.*

Bottleneck Analysis:

$$X(0) \leq \min\left[\frac{1}{max\ Di}, \frac{N}{\sum_{i=1}^{K} Di}\right]$$

The first throughput bond will be limited by the Maximum Service demand of a device within the System. The service demand from highest to lowest: *D(Disk1) > D(Disk3) > D(CPU) > Disk (Disk2).* So, the first throughput bound value will be:

$$\frac{1}{Max\ Di} = \frac{1}{3.698} = 0.27042\ (\text{jobs/s})$$

The Second bond, N is the number of Interactive Users and sums the service demand of all devices. Additionally, the throughput bound will also affect by the Thinking time:

$$\frac{N}{\sum_{i=i}^{K} Di + Thinking\ Time} = \frac{N}{D(Disk1) + D(Disk2) + D(Disk3) + D(CPU) + Thinking\ Time}$$
$$= \frac{4}{3.698 + 3.447 + 3.635 + 3.379 + 20} = \frac{4}{34.1159} = 0.11725\ (jobs/s)$$

Thus, by using the Bottleneck Analysis to get the Asymptotic bound:

$$X(0) \leq \min\left[\frac{1}{max\ Di}, \frac{N}{\sum_{i=1}^{K} Di}\right] = min[0.27042, \qquad 0.11725] = 0.11725$$

Question 2 (7 Marks)

A call centre has 2 staff to deal with customer enquires. The centre has a dispatcher to direct the calls automatically to one of the staff. The dispatcher does not contain any queueing facilities. At each staff's terminal, there is a facility to queue up to 3 calls. The queueing network at the call centre is depicted in Figure 1.

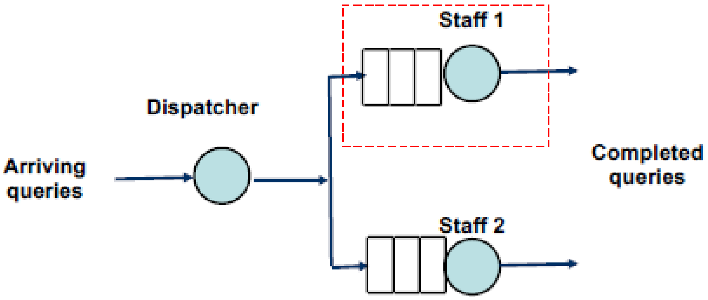


Figure 1: Depiction of the call centre.

The centre receives on average λ queries per hour. The arrivals can be modelled by using the Poisson distribution.

When a query arrives at the dispatcher, it will send the query to Staff 1 with a probability of p and to Staff 2 with a probability of $1-p$. Note that the dispatcher does not communicate with the staff's terminals, so it is possible that the dispatcher sends a query to a terminal that has a full queue. You can assume that the dispatcher takes a negligible time to perform its work and no queries will be dropped at the dispatcher.

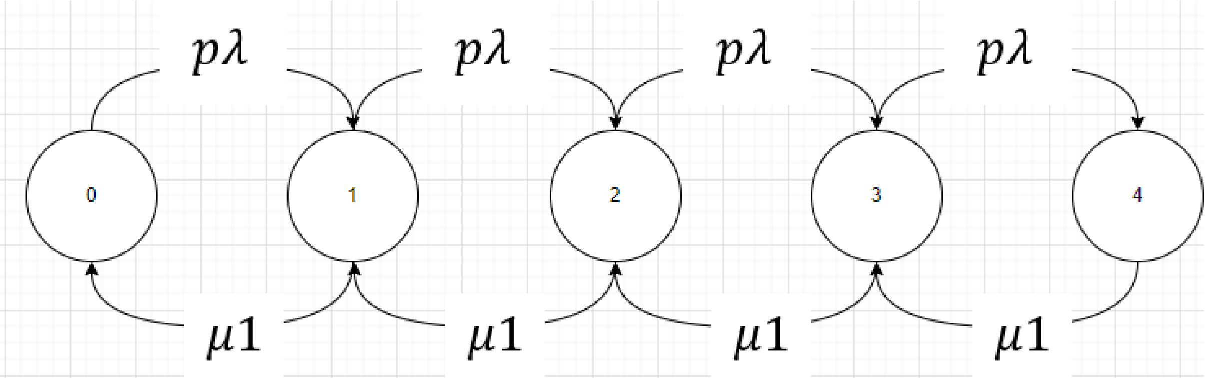
Staff 1 and Staff 2 can complete, respectively, on average μ_1 and μ_2 queries per hour. The amount of time required by each staff is exponentially distributed.

When a query arrives at a staff's terminal, it will be answered straight away if the staff is not busy. Otherwise, the terminal will place the call in its queue if the queue is not full. If the call arrives when the queue is full, then the call is rejected.

Answer the following questions:

a) ***Formulate a continuous-time Markov chain for the part of the call centre consisting of <u>Staff 1 and their three waiting slots</u>***

The continuous-time Markov Chain for the part of the Call centre Consisting of Staff one described as Below:



Brief explanation of the term:

- p: The probability of Call that assign to Staff 1 from Dispatcher.
- λ: The Centre receives on average queries per hour.
- μ_1: Staff 1 complete average queries per hour.
- $p\lambda$: $p * \lambda$ in result can calculate the queries that assign to Staff 1.

Define the States:

- State 0: Staff 1 is Idle and waiting for calls.
- State 1: Staff 1 receives a call and serve the query right away.
- State 2: Staff 1 is serving a call, one query in waiting slots.
- State 3: Staff 1 is serving a call, two queries in waiting slots.
- State 4: Staff 1 is serving a call, three queries in waiting slots and the slots are full. A further query is assigned to Staff 1 will be rejected.

b) ***Write down the balance equations for the continuous-time Markov chain that you have formulated.***

Brief Explanation of the Terms:

- P_i: Probability in State i.

Balance Equation List:

- $p\lambda P_0 = \mu_1 P_1$
- $p\lambda P_1 = \mu_1 P_2$
- $p\lambda P_2 = \mu_1 P_3$
- $p\lambda P_3 = \mu_1 P_4$
- $p\lambda P_0 + \mu_1 P_2 = (\mu_1 + p\lambda)\ P_1$
- $p\lambda P_1 + \mu_1 P_3 = (\mu_1 + p\lambda)\ P_2$
- $p\lambda P_2 + \mu_1 P_4 = (\mu_1 + p\lambda)\ P_3$

c) Derive the expressions for the steady state probabilities of the continuous-time Markov chain that you have formulated.

$$P_0 + P_1 + P_2 + P_3 + P_4 = 1$$

Steady State for P_0:

$p\lambda P_0 = \mu_1 P_1$ => $P_1 = \frac{p\lambda}{\mu 1} P_0$	$p\lambda P_1 = \mu_1 P_2$ => $P_2 = \frac{p\lambda}{\mu 1} P_1$ => $P_2 = \left(\frac{p\lambda}{\mu 1}\right)^2 * P_0$	$p\lambda P_2 = \mu_1 P_3$ => $P_3 = \frac{p\lambda}{\mu 1} P_2$ => $P_3 = \left(\frac{p\lambda}{\mu 1}\right)^3 * P_0$	$p\lambda P_3 = \mu_1 P_4$ => $P_4 = \frac{p\lambda}{\mu 1} P_3$ => $P_2 = \left(\frac{p\lambda}{\mu 1}\right)^4 * P_0$

$$P_0 + \frac{p\lambda}{\mu 1} * P_0 + \left(\frac{p\lambda}{\mu 1}\right)^2 * P_0 + \left(\frac{p\lambda}{\mu 1}\right)^3 * P_0 + \left(\frac{p\lambda}{\mu 1}\right)^4 * P_0 = 1$$

$$P_0\left(1 + \frac{p\lambda}{\mu 1} + \left(\frac{p\lambda}{\mu 1}\right)^2 + \left(\frac{p\lambda}{\mu 1}\right)^3 + \left(\frac{p\lambda}{\mu 1}\right)^4\right) = 1$$

$$P_0 = \frac{1}{1+\frac{p\lambda}{\mu 1}+\left(\frac{p\lambda}{\mu 1}\right)^2+\left(\frac{p\lambda}{\mu 1}\right)^3+\left(\frac{p\lambda}{\mu 1}\right)^4}$$

Steady State for P_1:

$p\lambda P_0 = \mu_1 P_1$ => $P_0 = \frac{\mu 1}{p\lambda} P_1$	$p\lambda P_1 = \mu_1 P_2$ => $P_2 = \frac{p\lambda}{\mu 1} * P_1$	$p\lambda P_2 = \mu_1 P_3$ => $P_3 = \frac{p\lambda}{\mu 1} P_2$ => $P_3 = \left(\frac{p\lambda}{\mu 1}\right)^2 * P_1$	$p\lambda P_3 = \mu_1 P_4$ => $P_4 = \frac{p\lambda}{\mu 1} P_3$ => $P_2 = \left(\frac{p\lambda}{\mu 1}\right)^3 * P_1$

$$\frac{\mu 1}{p\lambda} * P_1 + P_1 + \frac{p\lambda}{\mu 1} P_1 + \left(\frac{p\lambda}{\mu 1}\right)^2 * P_1 + \left(\frac{p\lambda}{\mu 1}\right)^3 * P_1 = 1$$

$$P_1\left(\frac{\mu 1}{p\lambda} + 1 + \frac{p\lambda}{\mu 1} + \left(\frac{p\lambda}{\mu 1}\right)^2 + \left(\frac{p\lambda}{\mu 1}\right)^3\right) = 1$$

$$P_1 = \frac{1 * \frac{p\lambda}{\mu 1}}{\left(\frac{\mu 1}{p\lambda}+1+\frac{p\lambda}{\mu 1}+\left(\frac{p\lambda}{\mu 1}\right)^2+\left(\frac{p\lambda}{\mu 1}\right)^3\right) * \frac{p\lambda}{\mu 1}}$$

$$P_1 = \frac{\frac{p\lambda}{\mu 1}}{1+\frac{p\lambda}{\mu 1}+\left(\frac{p\lambda}{\mu 1}\right)^2+\left(\frac{p\lambda}{\mu 1}\right)^3+\left(\frac{p\lambda}{\mu 1}\right)^4}$$

Steady State for P₂:

$p\lambda P_0 = \mu_1 P_1$ => $P_0 = \frac{\mu 1}{p\lambda} P_1$ => $P_0 = (\frac{\mu 1}{p\lambda})^2 P_2$	$p\lambda P_1 = \mu_1 P_2$ => $P_1 = \frac{\mu 1}{p\lambda} * P_2$	$p\lambda P_2 = \mu_1 P_3$ => $P_3 = \frac{p\lambda}{\mu 1} P_2$	$p\lambda P_3 = \mu_1 P_4$ => $P_4 = \frac{p\lambda}{\mu 1} P_3$ => $P_2 = \left(\frac{p\lambda}{\mu 1}\right)^2 * P_2$

$$(\frac{\mu 1}{p\lambda})^2 P_2 + \frac{\mu 1}{p\lambda} * P_2 + P_2 + \frac{p\lambda}{\mu 1} P_2 + \left(\frac{p\lambda}{\mu 1}\right)^2 * P_2 = 1$$

$$P_2 \left((\frac{\mu 1}{p\lambda})^2 + \frac{\mu 1}{p\lambda} + 1 + \frac{p\lambda}{\mu 1} + \left(\frac{p\lambda}{\mu 1}\right)^2\right) = 1$$

$$P_2 = \frac{1 * (\frac{p\lambda}{\mu 1})^2}{\left((\frac{\mu 1}{p\lambda})^2 + \frac{\mu 1}{p\lambda} + 1 + \frac{p\lambda}{\mu 1} + (\frac{p\lambda}{\mu 1})^2\right) * (\frac{p\lambda}{\mu 1})^2}$$

$$P_2 = \frac{(\frac{p\lambda}{\mu 1})^2}{1 + \frac{p\lambda}{\mu 1} + (\frac{p\lambda}{\mu 1})^2 + (\frac{p\lambda}{\mu 1})^3 + (\frac{p\lambda}{\mu 1})^4}$$

Steady State for P₃:

$p\lambda P_0 = \mu_1 P_1$ => $P_0 = \frac{\mu 1}{p\lambda} P_1$ => $P_0 = (\frac{\mu 1}{p\lambda})^3 P_3$	$p\lambda P_1 = \mu_1 P_2$ => $P_1 = (\frac{\mu 1}{p\lambda})^2 * P_3$	$p\lambda P_2 = \mu_1 P_3$ => $P_2 = \frac{\mu 1}{p\lambda} * P_3$	$p\lambda P_3 = \mu_1 P_4$ => $P_4 = \frac{p\lambda}{\mu 1} P_3$ => $P_4 = \left(\frac{p\lambda}{\mu 1}\right)^1 * P_3$

$$(\frac{\mu 1}{p\lambda})^3 P_3 + (\frac{\mu 1}{p\lambda})^2 * P_3 + \frac{\mu 1}{p\lambda} * P_3 + P_3 + (\frac{p\lambda}{\mu 1})^1 * P_3 = 1$$

$$P_3 \left((\frac{\mu 1}{p\lambda})^3 + (\frac{\mu 1}{p\lambda})^2 + \frac{\mu 1}{p\lambda} + 1 + (\frac{p\lambda}{\mu 1})^1\right) = 1$$

$$P_3 = \frac{1 * (\frac{p\lambda}{\mu 1})^3}{((\frac{\mu 1}{p\lambda})^3 + (\frac{\mu 1}{p\lambda})^2 + \frac{\mu 1}{p\lambda} + 1 + (\frac{p\lambda}{\mu 1})^1)) * (\frac{p\lambda}{\mu 1})^3}$$

$$P_3 = \frac{(\frac{p\lambda}{\mu 1})^3}{1 + \frac{p\lambda}{\mu 1} + (\frac{p\lambda}{\mu 1})^2 + (\frac{p\lambda}{\mu 1})^3 + (\frac{p\lambda}{\mu 1})^4)}$$

Steady State for P₄:

$p\lambda P_0 = \mu_1 P_1$ => $P_0 = \frac{\mu 1}{p\lambda} P_1$ => $P_0 = (\frac{\mu 1}{p\lambda})^4 P_4$	$p\lambda P_1 = \mu_1 P_2$ => $P_1 = (\frac{\mu 1}{p\lambda})^3 * P_4$	$p\lambda P_2 = \mu_1 P_3$ => $P_2 = (\frac{\mu 1}{p\lambda})^2 * P_4$	$p\lambda P_3 = \mu_1 P_4$ => $P_3 = \frac{\mu 1}{p\lambda} P_4$ => $P_3 = \frac{\mu 1}{p\lambda} * P_4$

$$(\frac{\mu 1}{p\lambda})^4 P_4 + (\frac{\mu 1}{p\lambda})^3 * P_4 + (\frac{\mu 1}{p\lambda})^2 * P_4 + \frac{\mu 1}{p\lambda} * P_4 + P_4 = 1$$

$$P_4 = \frac{1 * (\frac{p\lambda}{\mu 1})^4}{((\frac{\mu 1}{p\lambda})^4 + (\frac{\mu 1}{p\lambda})^3 + (\frac{\mu 1}{p\lambda})^2 + \frac{\mu 1}{p\lambda} + 1) * (\frac{p\lambda}{\mu 1})^4}$$

$$P_4 = \frac{(\frac{p\lambda}{\mu 1})^4}{1 + \frac{p\lambda}{\mu 1} + (\frac{p\lambda}{\mu 1})^2 + (\frac{p\lambda}{\mu 1})^3 + (\frac{p\lambda}{\mu 1})^4}$$

d) Assuming that p = 0.4, λ = 5.7 and μ 1 = 6.1. Determine the probability that a query that is dispatched to Staff 1 will be rejected.

Query from Dispatched to Staff 1 get rejected mean that Staff 1 must be in State 4 as other state will have Waiting slot for query.

$$P_4 = \frac{\left(\frac{p\lambda}{\mu 1}\right)^4}{1+\frac{p\lambda}{\mu 1}+\left(\frac{p\lambda}{\mu 1}\right)^2+\left(\frac{p\lambda}{\mu 1}\right)^3+\left(\frac{p\lambda}{\mu 1}\right)^4} = \frac{\left(\frac{0.4*5.7}{6.1}\right)^4}{1+\left(\frac{0.4*5.7}{6.1}\right)+\left(\frac{0.4*5.7}{6.1}\right)^2+\left(\frac{0.4*5.7}{6.1}\right)^3+\left(\frac{0.4*5.7}{6.1}\right)^4} \approx 0.0123$$

e) Assuming that p = 0.4, λ = 5.7, μ 1 = 6.1 and μ 2 = 6.5, determine the mean waiting time of the queries that have not been rejected by the call centre. Note that Part (d) considers only queries that have been dispatched to Sta_ 1 but Part (e) considers the whole call centre.

Staff 1 (Steady State Calculation):

P₀ = $\frac{1}{1+\frac{p\lambda}{\mu 1}+\left(\frac{p\lambda}{\mu 1}\right)^2+\left(\frac{p\lambda}{\mu 1}\right)^3+\left(\frac{p\lambda}{\mu 1}\right)^4} \approx 0.63083$	**P₁ =** $\frac{\frac{p\lambda}{\mu 1}}{1+\frac{p\lambda}{\mu 1}+\left(\frac{p\lambda}{\mu 1}\right)^2+\left(\frac{p\lambda}{\mu 1}\right)^3+\left(\frac{p\lambda}{\mu 1}\right)^4} \approx 0.23578$
P₂ = $\frac{(\frac{p\lambda}{\mu 1})^2}{1+\frac{p\lambda}{\mu 1}+\left(\frac{p\lambda}{\mu 1}\right)^2+\left(\frac{p\lambda}{\mu 1}\right)^3+\left(\frac{p\lambda}{\mu 1}\right)^4} \approx 0.08812$	**P₃ =** $\frac{(\frac{p\lambda}{\mu 1})^3}{1+\frac{p\lambda}{\mu 1}+\left(\frac{p\lambda}{\mu 1}\right)^2+\left(\frac{p\lambda}{\mu 1}\right)^3+\left(\frac{p\lambda}{\mu 1}\right)^4} \approx 0.03294$
P₄ = $\frac{(\frac{p\lambda}{\mu 1})^4}{1+\frac{p\lambda}{\mu 1}+\left(\frac{p\lambda}{\mu 1}\right)^2+\left(\frac{p\lambda}{\mu 1}\right)^3+\left(\frac{p\lambda}{\mu 1}\right)^4} \approx 0.01231$	

Since any request that arrives at stage 4 will be rejected, so we exclude it for the mean number of jobs calculation.

N1 = $\sum_{k=0}^{3} kP_k$ = $0 * 0.63083 + 1 * 0.23578 + 2 * 0.08812 + 3 * 0.03294 \approx$ 0.51084

According to Little's Law: Mean number of Job = Throughput x Response Time

Response Time (Staff 1) = N1/ $p\lambda$ = 0.51084/ (0.4 * 5.7) = 0.22405

Mean waiting time = Mean Response Time – Mean Service Time

Waiting time (Staff 1) = 0.22405 – 1/6.5= 0.07020

Staff 2 (Steady State Calculation):

P$_0$ = $\frac{1}{1+\frac{(1-p)\lambda}{\mu2}+\left(\frac{(1-p)\lambda}{\mu2}\right)^2+\left(\frac{(1-p)\lambda}{\mu2}\right)^3+\left(\frac{(1-p)\lambda}{\mu2}\right)^4} \approx 0.49375$	**P$_1$ =** $\frac{\frac{(1-p)\lambda}{\mu2}}{1+\frac{(1-p)\lambda}{\mu2}+\left(\frac{(1-p)\lambda}{\mu2}\right)^2+\left(\frac{(1-p)\lambda}{\mu2}\right)^3+\left(\frac{(1-p)\lambda}{\mu2}\right)^4} \approx 0.25979$
P$_2$ = $\frac{\left(\frac{(1-p)\lambda}{\mu2}\right)^2}{1+\frac{(1-p)\lambda}{\mu2}+\left(\frac{(1-p)\lambda}{\mu2}\right)^2+\left(\frac{(1-p)\lambda}{\mu2}\right)^3+\left(\frac{(1-p)\lambda}{\mu2}\right)^4} \approx 0.13669$	**P$_3$ =** $\frac{\left(\frac{(1-p)\lambda}{\mu2}\right)^3}{1+\frac{(1-p)\lambda}{\mu2}+\left(\frac{(1-p)\lambda}{\mu2}\right)^2+\left(\frac{(1-p)\lambda}{\mu2}\right)^3+\left(\frac{(1-p)\lambda}{\mu2}\right)^4} \approx 0.07192$
P$_4$ = $\frac{\left(\frac{(1-p)\lambda}{\mu2}\right)^4}{1+\frac{(1-p)\lambda}{\mu2}+\left(\frac{(1-p)\lambda}{\mu2}\right)^2+\left(\frac{(1-p)\lambda}{\mu2}\right)^3+\left(\frac{(1-p)\lambda}{\mu2}\right)^4} \approx 0.03784$	

N1 = $\sum_{k=0}^{3} kP_k$ = $0 * 0.49375 + 1 * \ 0.25979 + 2 * \ 0.13669 + 3 * \ 0.07192 \approx$ 0.74893

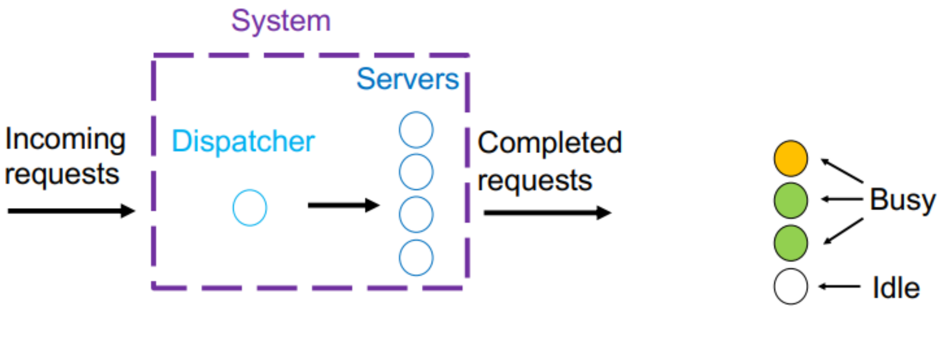
Response Time (Staff 2) = N1/ $(1-p)\lambda$ = 0. 74893/ (0.6 * 5.7) = 0.21898

Waiting time (Staff 2) = 0.21898 – 1/6.1 = 0.06011

Mean Waiting time (Whole Call Centre & No Reject Call)
= Mean Waiting time (Staff 1) * p + Mean Waiting time (Staff 2) * $(1-p)$
= 0.07020*0.4 + 0.06011 * 0.6
= 0.064146 (hour)

Question 3:

This question is based on the system illustrated in Figure 2a. The system consists of a dispatcher at the front-end and n servers at the back-end. We will use the value of $n = 4$ for explanation in Figures 2-4 but you will need to vary the value of n later on when answering the questions. Note that this system has no queueing facilities at neither the dispatcher nor the servers.



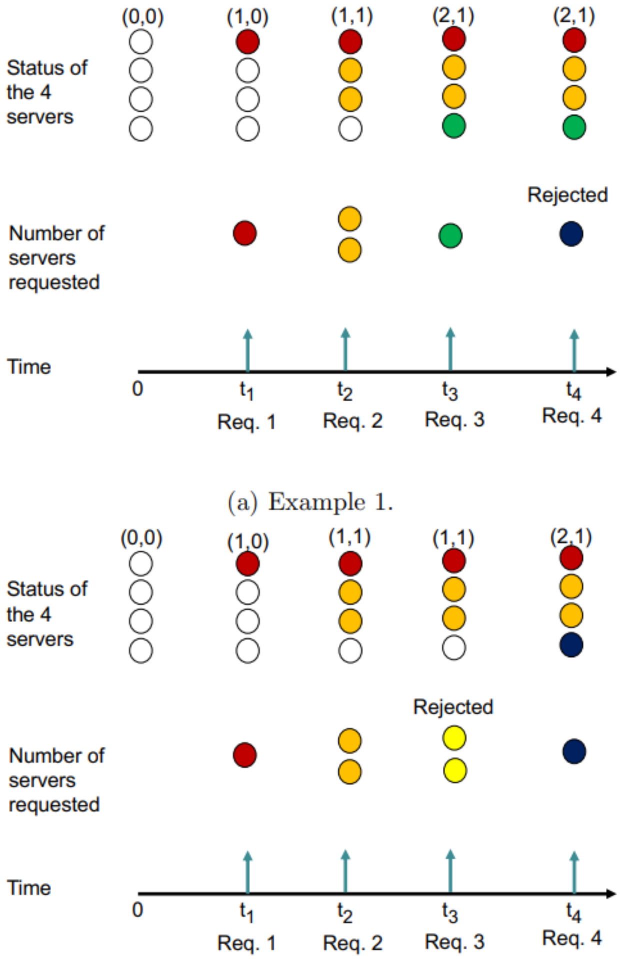
(a) (b)

Figure 2: (a) Pictorial representation of the system. (b) Conventions in Figures 3a and 3b: an unfilled circle means the server is idle and a filled circle means the server is ~~idle~~ busy.

This system is used to serve two classes of requests, which we will refer to as Class 1 and Class 2. Each request from Class 1 requires 1 server for processing while each request for Class 2 requires 2 servers for processing.

A function of the dispatcher is to decide whether an arriving request can be admitted or otherwise rejected. If there is at least an idle server at the time that a Class 1 request arrives, then the request will be admitted and dispatched to any one of the idle servers. If there are at least 2 idle servers at the time that a Class 2 request arrives, then the request will be admitted and dispatched to any two of the idle servers. If there are insufficient idle servers to admit a request, then that request will be rejected.

Figures 3a and 3b illustrates the admission and rejection of requests. These illustrations assume that all the servers are idle at time 0. Also, we assume that the processing of the admitted requests will not have been completed by the arrival time of Request 4, which means there are no departures in the time period illustrated in these two figures. The number of servers wanted by each request is shown. If there are sufficient number of idle servers available at the time the request arrives, the request will be admitted. Note that for the server status, an unfilled circle means a server is idle and a filled circle means a server is busy, see Figure 2b for the convention. Note that the server status shown is for the time after the admission decision of an arriving request has been made. As an example, in Figure 3a, Request 4 is rejected because there are no idle servers at the time that this request arrives. As another example, in Figure 3b, Request 3 is rejected because there is only one idle serve available at the time that this request arrives but this request wants two servers.



(a) Example 1.

(b) Example 2.

Figure 3: Illustrating the admission and rejection decisions.

You can assume that the dispatcher takes a negligible time to perform its function. This means that if a request is admitted, then its processing begins at its time of arrival because the dispatcher takes negligible time to admit a request and send it to the server(s).

For each request from Class 1, the amount of work needed to perform by a server is exponentially distributed with mean w_1. We use Figure 4 to explain the processing requirements of a Class 2 request. Figure 4 shows that, if a Class 2 request is admitted, then both servers allocated to process that request will start the processing at the same time and both servers will also complete the processing at the same time. The amount of work for Class 2 requests will be specified on a *per server* basis. We will assume that, for each request from Class 2, the amount of work needed to perform per server is exponentially distributed with mean w_2. Note that requests will leave the system once their processing is complete.

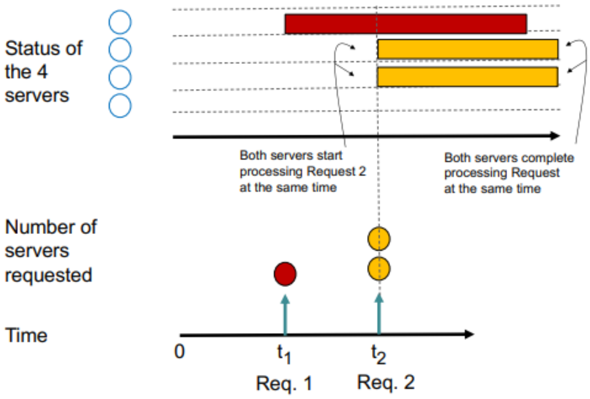

Status of the 4 servers
Both servers start processing Request 2 at the same time
Both servers complete processing Request at the same time
Number of servers requested
Time
0
t_1
t_2
Req. 1
Req. 2

Figure 4: The processing of a Class 2 request will start at the two allocated servers at the same time and also end at the same time.

The mean arrival rates of Class 1 and Class 2 requests are respectively, λ_1 and λ_2; and their arrival distributions are Poisson. Furthermore, you can assume that the arrivals from the two Classes are independent. The units for both λ_1 and λ_2 are number of requests per hour.

Each server has a constant processing rate of μ. The unit of μ is amount of work per hour. The units for w_1 and w_2 are the amount of work. You can assume that there is no processing overhead so the amount of time needed for processing is the amount of work divided by the processing rate.

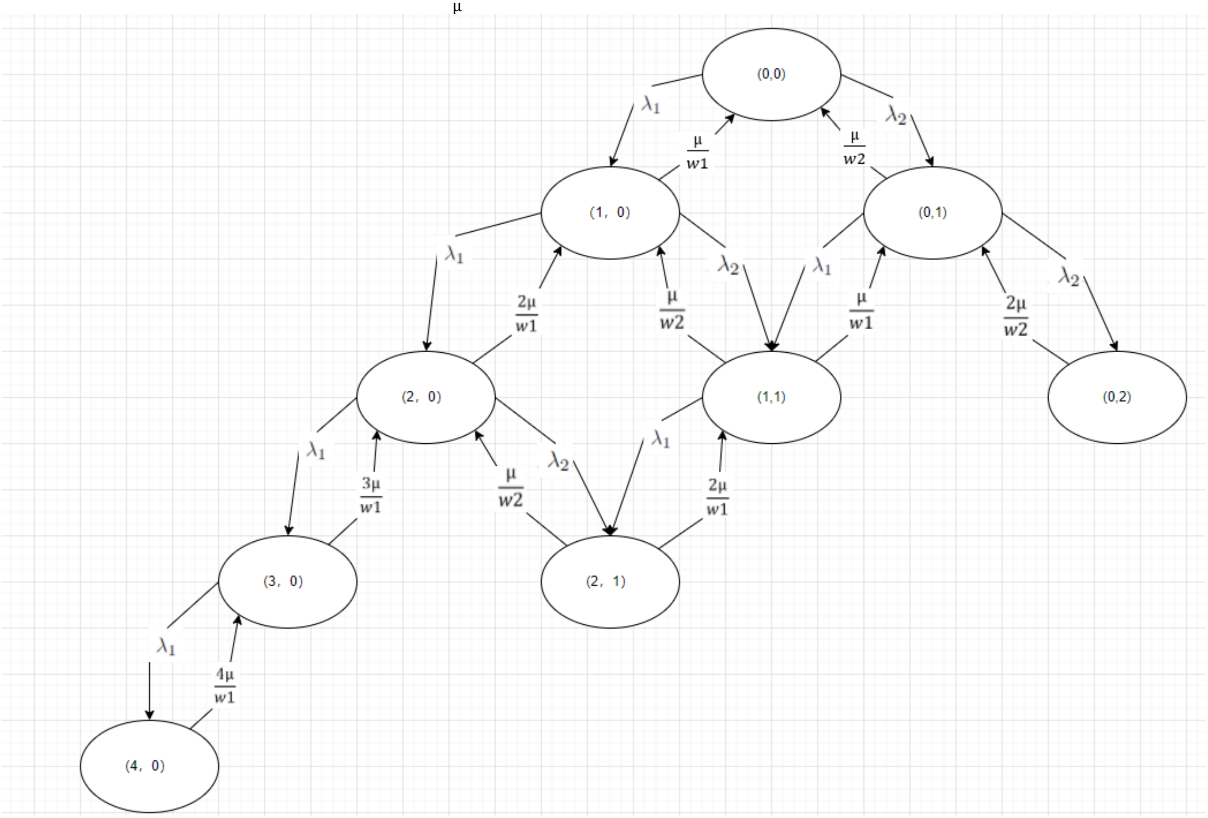
The system considered in this question can be modelled by a continuous-time Markov chain whose state is the tuple (r_1, r_2) where r_1 and r_2 are, respectively, the numbers of Class 1 and Class 2 requests in the servers. For example, in Figure 3a, the state of the system is $(2, 1)$ after Request 3 has been admitted. See Figures 3a and 3b for other examples.

Answer the following questions. Please note that you will be using a specific value of n for Parts (a) and (b), but you will need to vary n later on in Part (c).

The Following Question uses a Python Program for calculation.

a) Assuming that n = 4, formulate a continuous-time Markov chain for the system using the state definition given earlier. You can answer this question by drawing a state transition diagram with all the states and transitions. You can express the transition rates in terms of $\lambda 1$, $\lambda 2$, w1, w2 and µ.

w1: Workload of Class 1 query.
w2: Workload of Class 2 query.
µ: The Constant processing rate of a server (Hourly).
$\frac{w1}{\mu}$ & $\frac{w2}{\mu}$: The time that used to Complete the workload.
$\frac{\mu}{w1}$ & $\frac{\mu}{w2}$: $\frac{1\,(hr)}{\frac{w}{\mu}} = \frac{\mu}{w}$ is the processing rate work workload per hour.



- There are 4 Servers, which means this model is M/M/m.
- We treat Class 1 Request as a one-person team and Class 2 Request as a Two-people team. So, when a Request finishes, we return a probability of a Class team finish.
- When State (2,1) to (1,1). There are two Class 1 Requests, and we need to consider the probability of either one of the Class 1 requests finishing. So, we need to µ/w1 * 2.

b) Assuming that n = 4, λ_1 = 2.7, λ_2 = 1.5, w1 = 10.4, w2 = 15.3 and µ = 70. Answer the following questions.

(i) What are the steady state probabilities of the states for the continuous-time Markov chain?

$$(\lambda1\ +\ \lambda2)*\boldsymbol{P(0,0)}-\frac{\mu}{w1}*\boldsymbol{P(1,0)}-\frac{\mu}{w2}*\boldsymbol{P(0,1)}+0*\boldsymbol{P(2,0)}+\ 0*\boldsymbol{P(1,1)}+\ 0*\boldsymbol{P(0,2)}+\ 0*\boldsymbol{P(3,0)}+\ 0*\boldsymbol{P(2,1)}+\ 0*\boldsymbol{P(4,0)}=\mathbf{0}$$

$$-\lambda1*\boldsymbol{P(0,0)}+\left(\frac{\mu}{w1}+\lambda1+\lambda2\right)*\boldsymbol{P(1,0)}+0*\boldsymbol{P(0,1)}-\frac{2\mu}{w1}*\boldsymbol{P(2,0)}-\frac{\mu}{w2}*\boldsymbol{P(1,1)}+\ 0*\boldsymbol{P(0,2)}+\ 0*\boldsymbol{P(3,0)}+\ 0*\boldsymbol{P(2,1)}+\ 0*\boldsymbol{P(4,0)}=\mathbf{0}$$

$$-\lambda2*\boldsymbol{P(0,0)}+0*\boldsymbol{P(1,0)}+\left(\frac{\mu}{w2}+\lambda1+\lambda2\right)*\boldsymbol{P(0,1)}+0*\boldsymbol{P(2,0)}-\frac{\mu}{w1}*\boldsymbol{P(1,1)}-\frac{2\mu}{w2}*\boldsymbol{P(0,2)}+0*\boldsymbol{P(3,0)}+0*\boldsymbol{P(2,1)}+0*\boldsymbol{P(4,0)}=\mathbf{0}$$

$$0*\boldsymbol{P(0,0)}-\lambda1*\boldsymbol{P(1,0)}+0*\boldsymbol{P(0,1)}+\left(\frac{2\mu}{w1}+\lambda1+\lambda2\right)*\boldsymbol{P(2,0)}+0*\boldsymbol{P(1,1)}+0*\boldsymbol{P(0,2)}-\left(\frac{3\mu}{w1}\right)*\boldsymbol{P(3,0)}-\left(\frac{\mu}{w2}\right)*\boldsymbol{P(2,1)}+0*\boldsymbol{P(4,0)}=\mathbf{0}$$

$$0*\boldsymbol{P(0,0)}-\lambda2*\boldsymbol{P(1,0)}-\lambda1*\boldsymbol{P(0,1)}+0*\boldsymbol{P(2,0)}+\left(\frac{\mu}{w2}+\frac{\mu}{w1}+\lambda1\right)*\boldsymbol{P(1,1)}+0*\boldsymbol{P(0,2)}+0*\boldsymbol{P(3,0)}-(\frac{2\mu}{w1})*\boldsymbol{P(2,1)}+0*\boldsymbol{P(4,0)}=\mathbf{0}$$

$$0*\boldsymbol{P(0,0)}+0*\boldsymbol{P(1,0)}-\lambda2*\boldsymbol{P(0,1)}+0*\boldsymbol{P(2,0)}+0*\boldsymbol{P(1,1)}+\left(\frac{2\mu}{w2}\right)*\boldsymbol{P(0,2)}+0*\boldsymbol{P(3,0)}+0*\boldsymbol{P(2,1)}+0*\boldsymbol{P(4,0)}=\mathbf{0}$$

$$0*\boldsymbol{P(0,0)}+0*\boldsymbol{P(1,0)}+0*\boldsymbol{P(0,1)}-\lambda1*\boldsymbol{P(2,0)}+0*\boldsymbol{P(1,1)}+0*\boldsymbol{P(0,2)}+\left(\lambda1+\frac{3\mu}{w1}\right)*\boldsymbol{P(3,0)}+0*\boldsymbol{P(2,1)}-(\frac{4\mu}{w1})*\boldsymbol{P(4,0)}=\mathbf{0}$$

$$0*\boldsymbol{P(0,0)}+0*\boldsymbol{P(1,0)}+0*\boldsymbol{P(0,1)}-\lambda2*\boldsymbol{P(2,0)}-\lambda1*\boldsymbol{P(1,1)}+0*\boldsymbol{P(0,2)}+0*\boldsymbol{P(3,0)}+\left(\frac{\mu}{w2}+\frac{2\mu}{w1}\right)*\boldsymbol{P(2,1)}+0*\boldsymbol{P(4,0)}=\mathbf{0}$$

$$0*\boldsymbol{P(0,0)}+0*\boldsymbol{P(1,0)}+0*\boldsymbol{P(0,1)}+0*\boldsymbol{P(2,0)}+0*\boldsymbol{P(1,1)}+0*\boldsymbol{P(0,2)}-\lambda1*\boldsymbol{P(3,0)}+0*\boldsymbol{P(2,1)}+(\frac{4\mu}{w1})*\boldsymbol{P(4,0)}=\mathbf{0}$$

P (0,0):	0.4918992967968496	P (0,2):	0.026437202997055195
P (1,0):	0.19732188934365058	P (3,0):	0.005292028101012529
P (0,1):	0.16127269802125266	P (2,1):	0.012975645824598293
P (2,0):	0.03957713323406925	P (4,0):	0.0005307148181300137
P (1,1):	0.06469339086338233		

Please Check "Question3.py" and look for Q3_b_i

(ii) Determine the probability that an arriving Class 1 request will be rejected.

This Situation only happens when all servers are occupied.

P [Class 1 will be rejected]: = P(4,0) + P(2,1) + P(0,2)
= 0.039943563639783505

Please Check "Question3.py" and look for Q3_b_ii

(iii) Determine the probability that an arriving Class 2 request will be rejected.

This Situation only happens when less than 1 (Include 1) server is busy.

P [Class 2 will be rejected]: = P(1,1) + P(0,2) + P(3,0) + P(2,1) + P(4,0)
= 0.10992898260417834

Please Check "Question3.py" and look for Q3_b_iii

(iv) Determine the probability that an arriving request will be rejected. Note that the hint in Question 2 is applicable.

In order to calculate the Arriving request will be rejected. I can calculate probability of Class 1 income Request (λ_1) of an overall income Request ($\lambda_1 + \lambda_2$). The we can use the it calculate the probability of Class 1 arriving request will be rejected, same method apply to Class 2.

P [Class 1 Arriving Request rate] = $\frac{\lambda 1}{\lambda 1 + \lambda 2} = \frac{2.7}{2.7+1.5}$ = 0.642857

P [Arriving Request of Class 1 will be rejected] = $\frac{2.7}{2.7+1.5} *$ (P(4,0) + P(2,1) + P(0,2))

P [Arriving Request of Class 2 will be rejected] = $(1 - \frac{2.7}{2.7+1.5}) *$ (P(1,1) + P(0,2) + P(3,0) + P(2,1) + P(4,0))

P [Arriving Request will be rejected] =

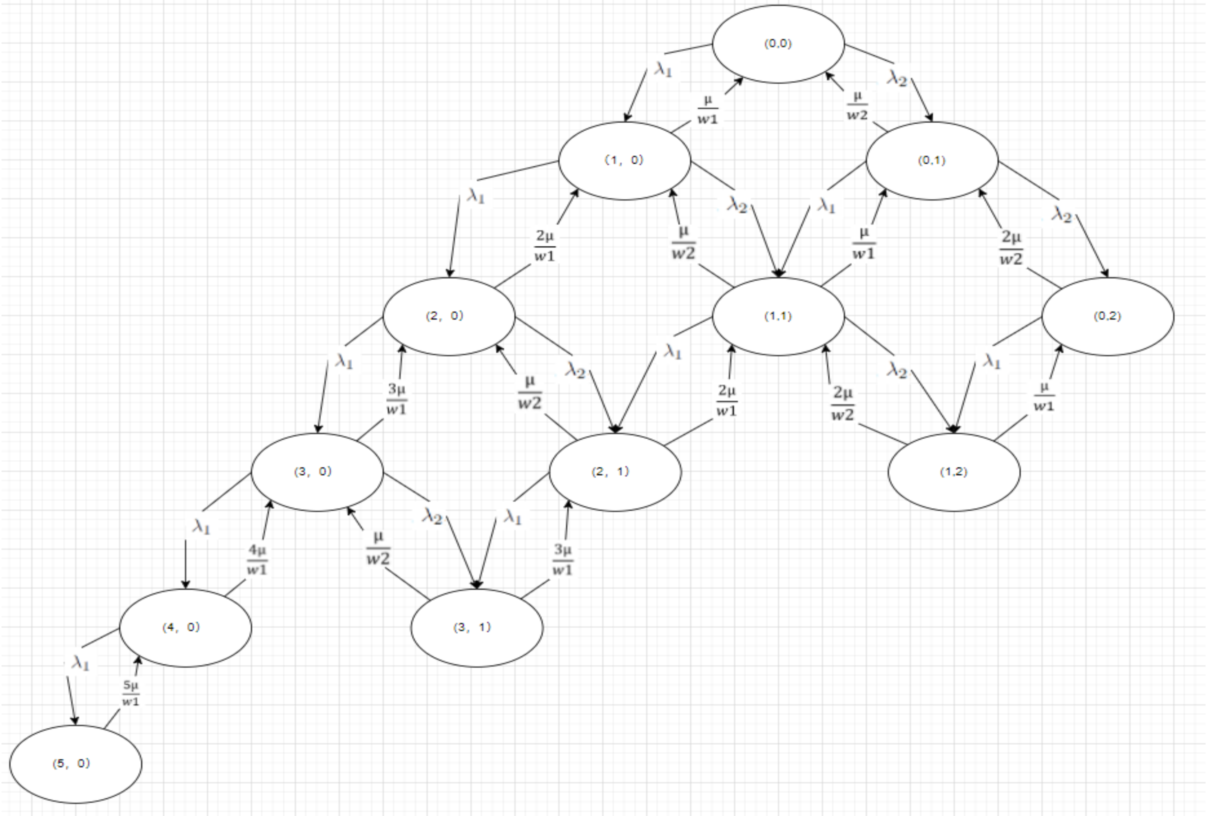
P [Arriving Request of Class 1 will be rejected] + P [Arriving Request of Class 2 will be rejected]

= 0.06493835612706737

Please Check "Question3.py" and look for Q3_b_iv

c) Assuming that $\lambda_1 = 2,7$, $\lambda_2 = 1,5$, $w_1 = 10.4$, $w_2 = 15.3$ and $\mu = 70$. What is the smallest value of n that can reduce the probability of rejecting an arriving request to a level lower than 0.05?

The overall request is being rejected is 0.0649 when there are 4 servers which are slightly higher than 0.05. We increase the number of servers to 5.



P (0,0):	0.48588275502151895	P (3,0):	0.005227299998424833
P (1,0):	0.19490839658577508	P (2,1):	0.012816937496137826
P (0,1):	0.1593001318249121	P (1,2):	0.010475381607420349
P (2,0):	0.03909305554377544	P (4,0):	0.0005242235141277194
P (1,1):	0.06390211002347906	P (3,1):	0.0017138076423407625
P (0,2):	0.026113843038440866	P (5,0):	4.20577036477519e-05

P (Class 1 will be Rejected) = 0.012231246953408863
P (Class 2 will be Rejected) = 0.05168625100211527б

$$P\ [\text{Class 1 Income Request}] = \frac{\lambda_1}{\lambda_1 + \lambda_2} = \frac{2.7}{2.7+1.5} = 0.642857$$

P [Arriving Request of Class 1 will be rejected] = P (Class 1 will be Rejected) * P [Class 1 Income Request]

P [Arriving Request of Class 2 will be rejected] = P (Class 2 will be Rejected) * (1 - P [Class 1 Income Request])

P [Arriving Request will be rejected]
= P [Arriving Request of Class 1 will be rejected] + P [Arriving Request of Class 2 will be rejected]
= 0.026322319827946868

So, we can conclude that when number of Server is 5, the probability of request will be rejected less than 0.05.

Please Check “Question3.py” and look for Q3_c