COMP933 – Programming of Computer Systems and Networks

Assignment 1 (Version 1.01), Term 1, 2023

Due 5:00pm, Fri 17 March 2023 (Friday Week 5)

Change log and version info

Updates, changes and clarifications will appear in this box.

- Version 1.01 (7 March 2023) revises the wording in Question 1.
- Version 1.00 issued on 27 February 2023

Instructions

(1) There are 3 questions in this assignment. Answer all questions.

(2) The total mark for this assignment is 20 marks.

(3) The submission deadline is 5:00pm Friday 17 March 2023. Submissions made after the deadline will incur a penalty of 5% per day. Late submissions will only be accepted until 5:00pm Wednesday 22 March 2023, after which no submissions will be accepted.

(4) In answering the questions, it is important for you to show your intermediate steps and state what arguments you have made to obtain the results. You need to note that both the intermediate steps and the arguments carry marks. Please note that we are **not** just interested in whether you can get the final numerical answer right, we are **more** interested to find out whether you understand the subject matter. We do that by looking at your intermediate steps and the arguments that you have made to obtain the answer. Thus, if you can show us the perfect intermediate steps and the in-between arguments but get the numerical values wrong for some reason, we will still award you marks for having understood the subject matter.

You can take a look at the solution to revision problems to get some ideas the level of explanation that is required.

(5) If you use a computer program to perform any part of your work, you **must** submit the program or you lose marks for the steps.

(6) This is an ind

(7) Your submissi

(a) A report on to the problems. This report can be typewritten or a scan s. This report must be in pdf format and must be named report.pdf. The submission system will only accept the name report.pdf.

(b) One or more computer programs if you use them to solve the problems numerically. You should use zip to archive all the computer programs into one file with the name supp.zip. The submission system will only accept this name. The report must refer to the programs so that we know which program is used for which part.

(8) Submission can be made via the course website.

(9) You can submit as many times as you wish before the deadline. A later submission will over-write the earlier one. We will only mark the last submission that you make.

(10) If you want to ask questions on the assignment, you can attend a consultation (see the Timetable section of the course website for dates and times) or post your question on the forum. Please note that if your forum post shows part of your solution or code, you must mark that forum post **private**.

(11) Additional assignment conditions:

- Joint work is not permitted on this assignment.
  - This is an individual assignment. The work you submit must be entirely your own work: submission of work even partly written by any other person is not permitted.
  - Do not request help from anyone other than the teaching staff of COMP9344.
  - Do not post your assignment work or code to the course forum.
  - Assignment submissions are routinely examined both automatically and manually for work written by others.

  Rationale: this assignment is designed to develop the individual skills needed to solve problems. Using work/code written by, or taken from, other people will stop you learning these skills. Other CSE courses focus on skills needed for working in a team.

- The use of AI generative tools, such as ChatGPT, is not permitted on this assignment.

  Rationale: this assignment is designed to develop your understanding of basic concepts. Using AI tools will stop you learning these fundamental concepts, which will significantly impact your ability to complete future courses. Moreover, ChatGPT has been found to give incorrect answers for advanced problems covered in this course.

- Sharing, publishing, or distributing your assignment work is not permitted.

  - Do not provide or show your assignment work to any other person, other than the teaching staff of COMP9334. For example, do not message your work to friend
  - Do n[illegible]gnment code via the Internet. For example, do not place[illegible] a public GitHub repository.

  Rationale: [illegible]aring your work, you are facilitating other students using yo[illegible]dents find your assignment work and submit part or all of [illegible] you may become involved in an academic integrity investiga[illegible]

- Sharing, publishing, or distributing your assignment work after the completion of COMP9334 is not permitted.
  - For example, do not place your assignment in a public GitHub repository after this offering of COMP9334 is over.

  Rationale: COMP9334 may reuse assignment themes covering similar concepts and content. If students in future terms find your assignment work and submit part or all of it as their own work, you may become involved in an academic integrity investigation.

Question 1 (5 marks)

Assuming that you [illegible] of an interactive computer system. The computer system consists of a [illegible] a disk. During an observation time of 1800 seconds, you obtained the fo[illegible]s from the system:

Average busy t[illegible]		1575 s
Disk busy time[illegible]		1124 s
Number of req[illegible]	[illegible]e computer system	57

This computer system is used by 16 interactive users and the thinking time per interactive user is 45 seconds.

You consider the current throughput of the system is too low. You are considering a proposal to upgrade the current CPU, which has 4 cores, to a new CPU with 6 cores and the same processing speed per core as that of the current CPU.

For this question, you can assume that the total workload remains the same before and after the upgrade. You can also assume that the ~~workload is~~ requests (note the plural) are almost evenly distributed among the cores at the moment and ~~the workload~~ requests can still be evenly distributed among the cores after the upgrade. As the system administrator, you know that when ~~the workload~~ each request (note the singular) uses the CPU, it uses only one core at a time, i.e., ~~the workload~~ a request does not use multiple cores concurrently.

Answer the following questions.

(a) Determine the current average service demand of a core.

(b) What will the average service demand per core be if the proposed CPU upgrade is carried out?
Hint: The service demand of a core depends on two factors: the number of visits to the core and the service time needed per visit to the core. For the set up of this question, one of these two factors remains the same after the upgrade, while the other factor will change.

(c) What will the throughput bound of the computer system be if the proposed CPU upgrade is carried out?

Reminder: If you use a computer program to derive your numerical answers, you ***must*** *include your computer program in your submission. Do not forget to show us your steps to obtain your answer.*

Question 2 (7 marks)

Assuming that you [illegible]U and you are happy for outsiders to use your CPU as long as these out[illegible]nterrupt yours and your work takes precedence over theirs. This questi[illegible]ple who donate their spare CPU time for scientific research.

In this question [illegible] *primary user* to refer to the CPU owner (which is you) and the term e[illegible]o the outsiders. We make the following assumptions:

- The CPU is configured as a single processing unit without any queueing spaces.
- If a request (which can be from the primary user or an external user) arrives when the CPU is idle, the request will be admitted to the CPU.
- If a request (which can be from the primary user or an external user) arrives when the CPU is working on a primary user's request, the arriving request will be rejected. This is because there are no queueing spaces in the CPU.
- If a request from the primary user arrives when the CPU is working on an external user's request, the external user's work will be terminated immediately and the primary user's request will be admitted to the CPU immediately. In this case, the external user loses their work and its remaining work will not be resumed. Therefore, you can consider that the external user has left permanently.
- If a request from an external user arrives when the CPU is working on another external user's request, the newly arriving request will be rejected.
- The inter-arrival times for the primary user's requests are exponentially distributed with mean arrival rate λ_p; those for the external users' requests are exponentially distributed with mean arrival rate λ_e.
- The service times of the primary user's requests are exponentially distributed with a mean service time of $\frac{1}{\mu_p}$; those for the external users' requests are exponentially distributed with mean $\frac{1}{\mu_e}$.
- The four probability distributions mentioned in the last two dot points are independent of each other.

Answer the following questions. You are expected to express your answers in terms of these rate parameters: λ_p, λ_e, μ_p and μ_e.

(a) Let us assume that at time t, you observe that the request at the CPU belongs to an external user. What is the probability that this observed request will still be in the CPU at time $(t + \Delta t)$ where Δt is an infinitesimal time change? You should express this probability in terms of Δt and any of the appropriate rate parameters.

(b) Formulate a continuous-time Markov chain for the CPU. Your formulation should include the definition of the states and the transition rates between states.

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

(d) Derive the exp[illegible]y state probabilities of the continuous-time Markov chain that yo[illegible] You should be able to solve for the steady state probabilities [illegible] answers in terms of λ_p, λ_e, μ_p and μ_e.

(e) What is the p[illegible]uest from the primary user will be admitted? Why is this probab[illegible]he rate parameters of the external users?

(f) What is the probability that a request from an external user will be admitted?

Question 3 (8 marks)

This question is bas[illegible]crated in Figure 1. The system consists of a database server and an exter[illegible]ase server consists of a front-end server and a back-end server; each se[illegible]ue. Each of the three queues in this system (i.e., external, front-end, [illegible]pacity to hold only one request.

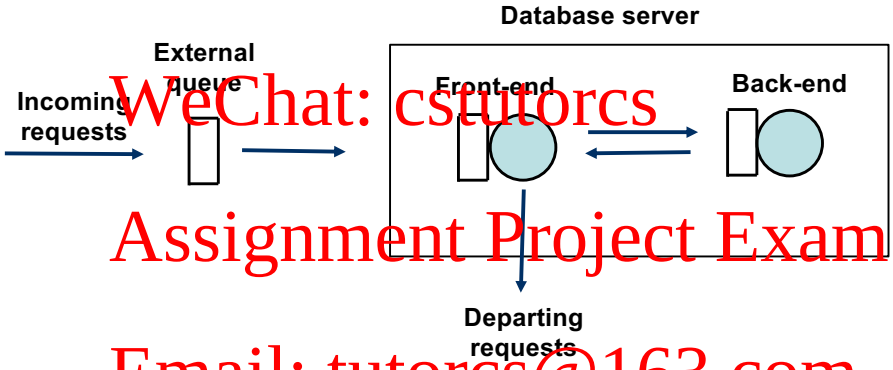

Figure 1

The mode of operation for the system in Figure 1 is as follows:

- The total number of requests in the database server (i.e., the two servers and two queues) must be two or less.
- If an incoming request arrives when there are a total of 2 requests in the database server, then the incoming request will join the external queue if it is empty; otherwise it will be rejected if the external queue is already occupied.
- If an incoming request arrives when there are no requests in the database server, then the incoming request will be sent to the front-end server.
- If there is one request in the database server, then an incoming request will be sent to the front-end server if it is idle or it will be placed in the front-end queue if the front-end server is busy.
- After the front-end server has finished processing a request, there is a probability of p that the front-end server will send the request to the back-end server for further processing, and a probability of $(1 - p)$ that the request will leave the database server (hence the system) permanently.
- After a request has been processed by the back-end server, the request will always be sent back to the front-end for further processing; this request will need to join the queue if the front-end server is busy.
- If there is a request waiting in the external queue at the time a request is leaving the database server permanently, then the request in the external queue will be admitted to

the database server. There are two scenarios depending on whether the front-end queue is occupied at the time when the permanent departure takes place. If the front-end queue is occu permanent departure, the request in the front-end queue will mc the request in the external queue will move to the front-end que ueue is unoccupied, then the request in the external queue will go r.

You can assume workload:

- The incoming distributed with a mean arrival rate of λ requests per unit time.
- The service time (i.e., per visit) to the front-end is exponentially distributed with a mean of $\frac{1}{\mu_f}$.
- The service time (i.e., per visit) to the back-end is exponentially distributed with a mean of $\frac{1}{\mu_b}$.
- All the service times and inter-arrival times are independent of each other.

(a) Formulate a continuous-time Markov chain for the system. Your formulation should include the definition of the states and the transition rates between states. The transition rates should be expressed in terms λ, μ_f, μ_b and p.

(b) Assuming that $\lambda = 1.4$, $\mu_f = 2.1$, $\mu_b = 1.8$ and $p = 0.3$.

(i) Determine the steady state probabilities of the state of the continuous-time Markov chain that you have specified in Part (a).

(ii) Determine the throughput of the database server.

(iii) Determine the mean response time of the database server.

Reminder: If you use a computer program to derive your numerical answers, you ***must*** *include your computer program in your submission. Do not forget to show us your steps to obtain your answer.*

– – –– End of assignment – – ––