**CS 2SD3 Midterm Test**

Monday, March 4, 2024, 7:30pm-9:30pm, Total: 62 pts

*Virtual Take Home Midterm Test*

● THIS EXAMINATION PAPER HAS **3** PAGES AND **7** QUESTIONS.

● The exam starts at 7:30am and ends at 8:50pm, i.e. 80 minutes (for students without extra time permissions). This includes the exam time (60 minutes) plus extra time for technology (20 minutes).

● Please submit the solutions via Avenue using the same procedure as for the assignments.

● Any question about the test questions, ask by sending an e-mail to Ryszard Janicki ([janicki@mcmaster.ca](mailto:janicki@mcmaster.ca)), while any question about Avenue ask by sending and e-mail to Sepehr Bayat ([bayats1@mcmaster.ca](mailto:bayats1@mcmaster.ca)).

1.[10] Consider the following scenario: An office is used by several workers sharing a single printer. As a simplification of a real life scenario it is assumed that the printer is able to print any number of jobs, independently of the number of pages those jobs have, before running out of toner. The toner is replaced by a technician whenever necessary.

a.[5] Provide an FSP description of the above scenario. You must provide also a brief description of the intended behavior for each one of the processes you define.

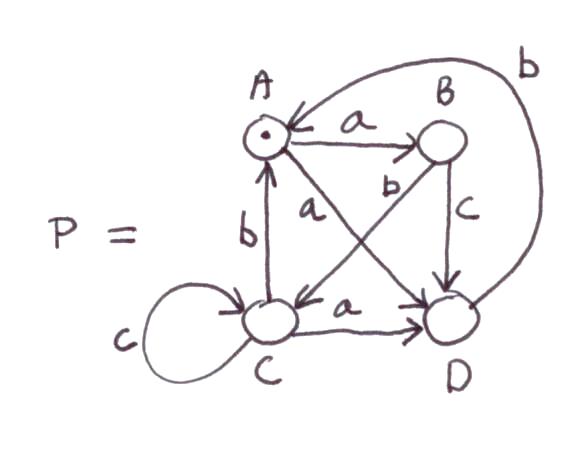
Hint: Possible processes for the above scenario are PRINTER, USER and TECHNICIAN.

b.[5] Provide a Petri nets (any kind) description of the above scenario.

2.[5] Consider the following simple hotel reservation system. A customer makes a room request. If room is available, a confirmation is sent to the customer, otherwise the customer is put on a reservation list. If a room is confirmed, the customer may either use it, pay for the room, leave and the whole transaction is archived. However the customer may also cancel his/her reservation. When the customer is on waiting list, a room may become available, and then a confirmation is sent to a customer. The customer may also give up waiting and cancel his/her request.

Model this reservation system as a FSP process reservation. Note that this process always stops, so you have to use the process STOP.

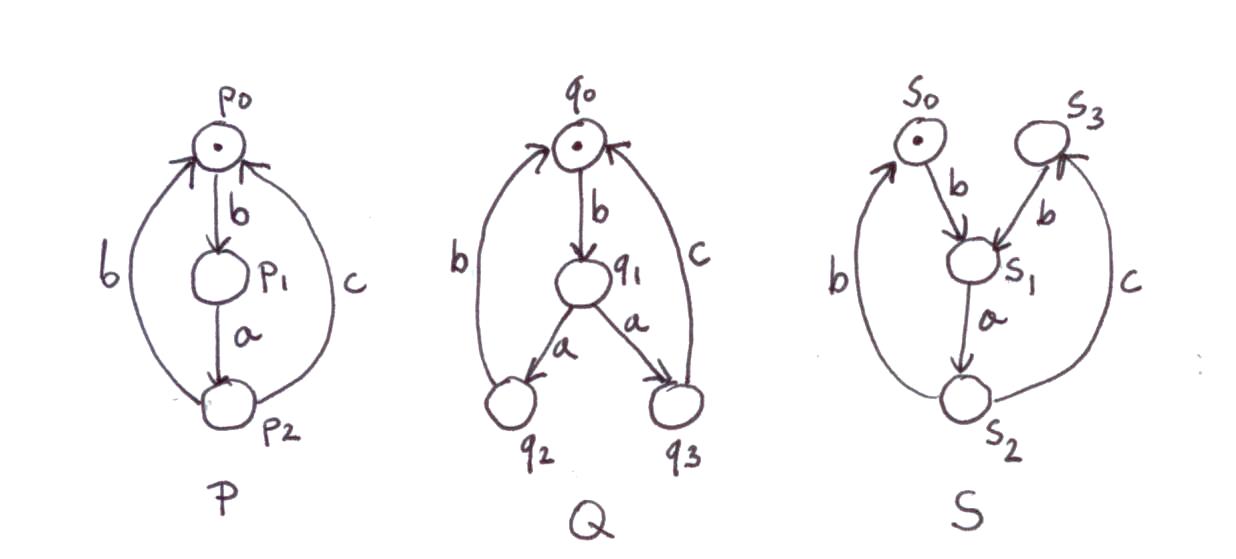
3.[5] Provide the Finite State Process (FSP) description of the following Labelled Transition Graph (dot indicate the initial state).



4.[12] Consider three Labelled Transition Systems given below: P, Q and S. Prove that:

a.[5] P ≉ Q, i.e. P and Q are *not bisimilar.*

b.[7] P ≈ S, i.e. P and S are *bisimilar*.



5.[10] Consider two philosophers system. Each philosopher either think, or eats cookies or drinks cola. However there is only one cookie dispenser and only one cola distributor (they are separate machines), so only one philosopher can get cookies or cola at a time, the other must wait.

a.[5] Model this system using FSP.

b.[5[ Model this system using any kind of Petri nets.

6.[14] Two workers W1 and W2 working separately need two different tools, say *drill* and *clamp* to do some work (say precise drill). In order to do the job, each worker needs get both tools. However, due to the nature of work, W1 needs to get drill first and clamp second, while W2 needs to get clamp first and drill second. We obviously want to avoid a situation when W1 grabs drill and waits for clamp, while W2 grabs clamp and waits for drill.

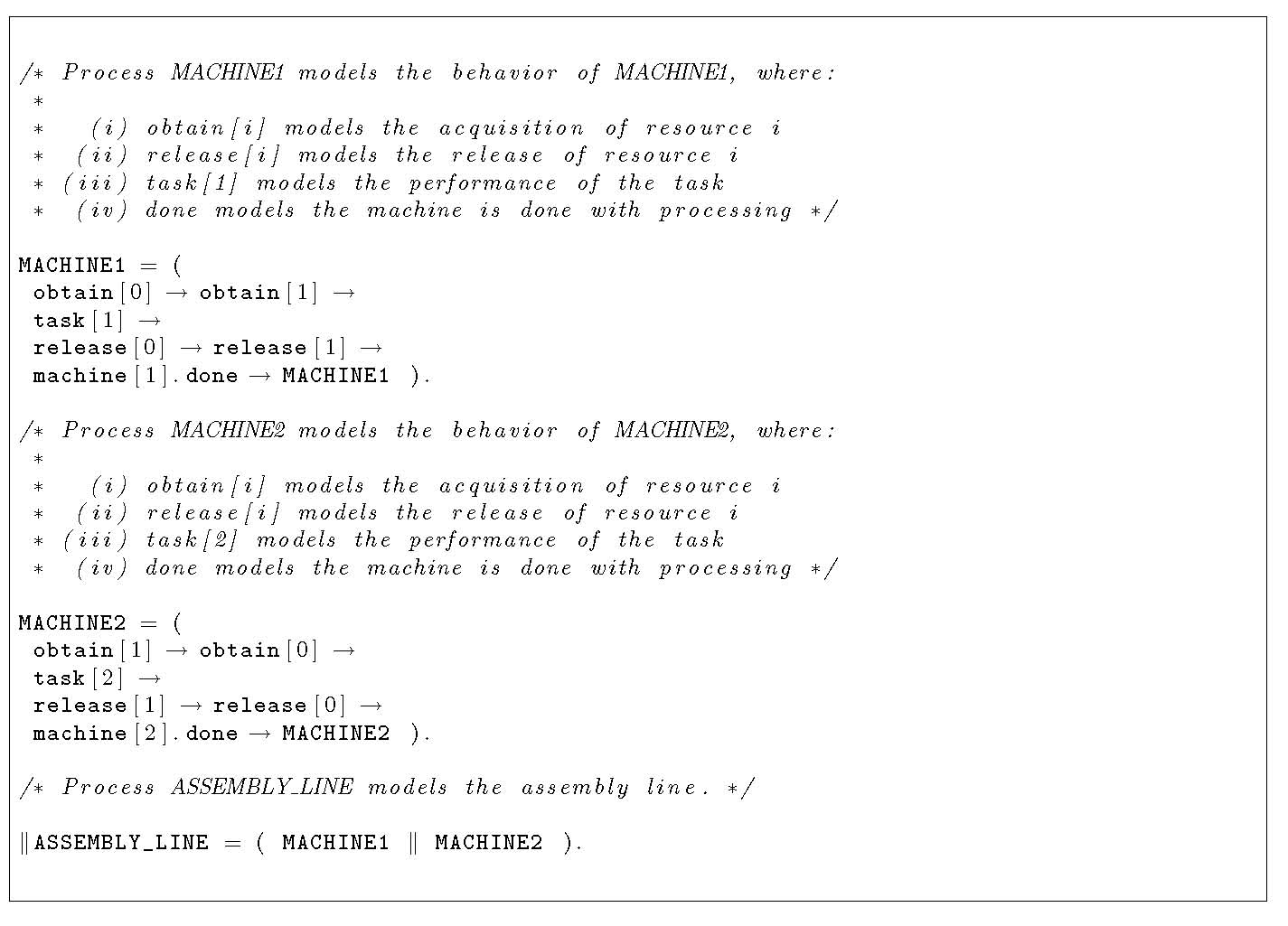
a.[7] Model the situation described above carefully avoiding deadlock with FSP.

b.[7] Model the situation described above carefully avoiding deadlock with Petri nets (any kind)

7.[6] In an assembly line in a factory machines perform several operations on different shared resources before completing a task. As the machines share different resources, it is of importance for an industrial engineer to determine the behavior of the machines interacting to avoid unwanted configurations, such as those leading to deadlock states.

Of special interest is the behavior of two machines operating on resources 0 and 1; as these must be used exclusively by one machine at a time. The behavior of the machines is as follows: to perform its task MACHINE1 must obtain resource 0 and then obtain resource 1. After performing it, MACHINE1 releases the acquired resources in the order in which they were acquired. In parallel, to perform its task, MACHINE2 must obtain resource 1 and then obtain resource 0. Analogously, MACHINE2 releases the resources in the order in which they were acquired.

Consider the following FSP description modeling the assembly line:



a.[2] Argue whether process ASSEMBLY\_LINE is a proper model for the assembly line.

b.[2] Is there a trace in ASSEMBLY\_LINE leading to a deadlock state.

c.[2] Is there a trace in ASSEMBLY\_LINE for which both machines are allowed to perform their tasks. That is, is there a trace t in the set of traces associated with process ASSEMBLY\_LINE such that the last two actions of t are *machine*[1].*done* and *machine*[2].*done*.