6SENG002W Concurrent Programming

FSP Process Analysis & Design Form

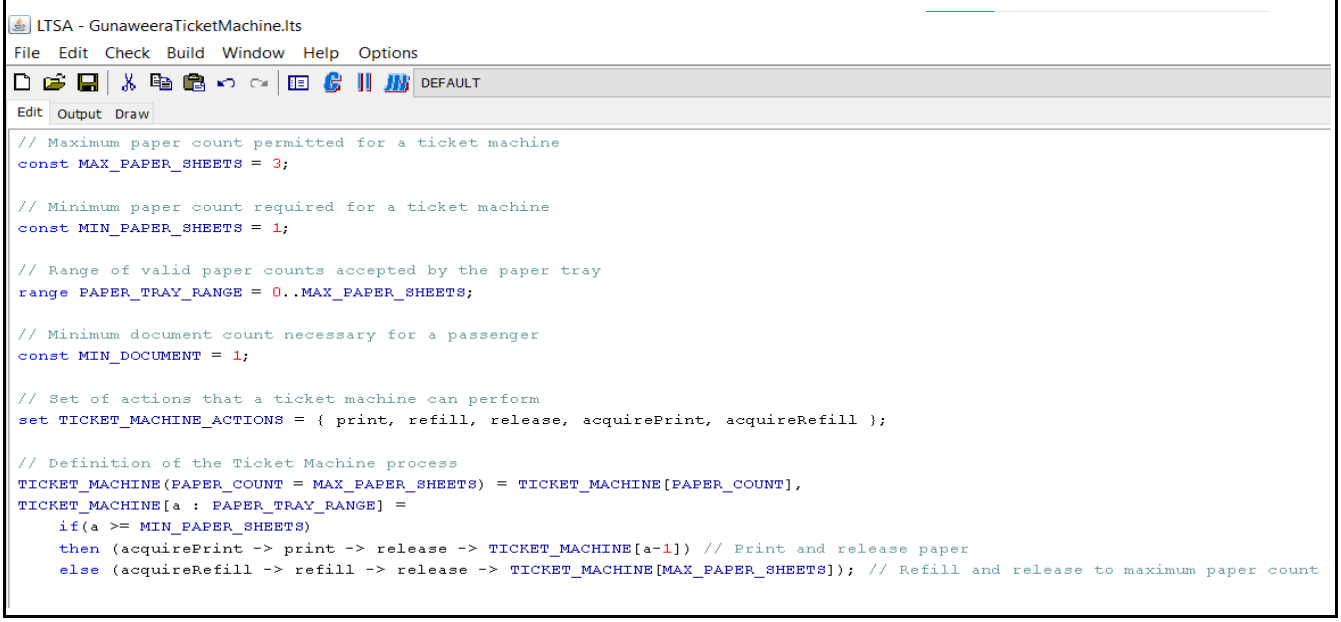
Name	H.K.J.N.Gunaweera
Student ID	IIT-20200003 UOW-w1810567
Date	31st of December 2023

1. FSP Process Attributes

Attribute	Value
Name	TICKET_MACHINE
Description	This is a model of a ticket machine, and its behaviour is based on printing ticket documents.Ticket machine will consume papers while printing ticket documents and its need to be refilled with paper. There is no deadlock or errors outperformed by the TICKETING_SYSTEM_COMPOSITION
Alphabet	alphabet(TICKET_MACHINE) = {acquirePrint, acquireRefill, print, refill, release}
Number of States	80
Deadlocks (yes/no)	No
Deadlock Trace(s) (if applicable)	Not applicable

2. FSP Process Code

FSP Process:



3. Actions Description

A description of what each of the FSP process' actions represents, i.e. is modelling. In addition, indicate if the action is intended to be synchronised (shared) with another process or asynchronous (not shared). (Add rows as necessary.)

Actions	Represents	Synchronous or Asynchronous
acquirePrint	Acquiring the lock on the ticket machine for mutual exclusive access to print a document.	Synchronous
acquireRefill	Acquiring the lock on the ticket machine for mutual exclusive access to refill the paper.	Synchronous
print	Print a ticket document	Synchronous
refill	Refilling the ticket machine with papers	Synchronous
release	Releasing the ticket machine after the document are printed or completing the refilling paper process.	Synchronous

4. FSM/LTS Diagrams of FSP Process

Note that if there are too many states, more than 64, then the LTSA tool will not be able to draw the diagram. In this case draw small diagrams of the most important parts of the complete diagram.

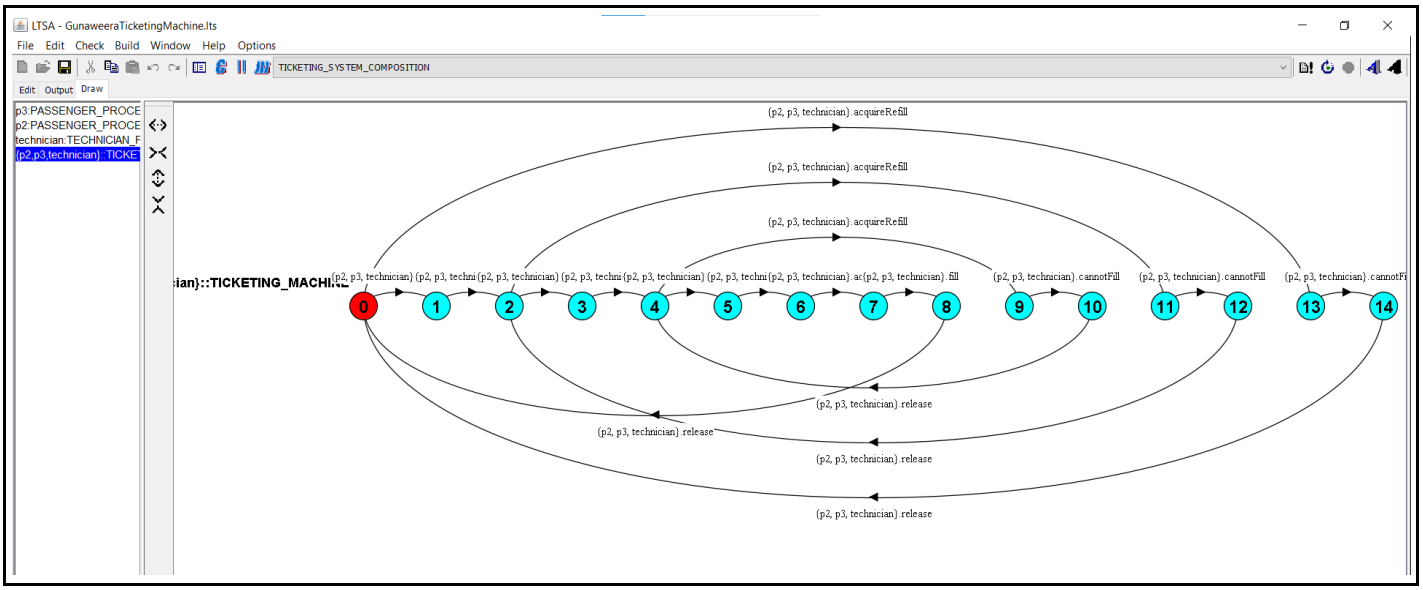


Figure 1- FSP process inside LTSA tool view

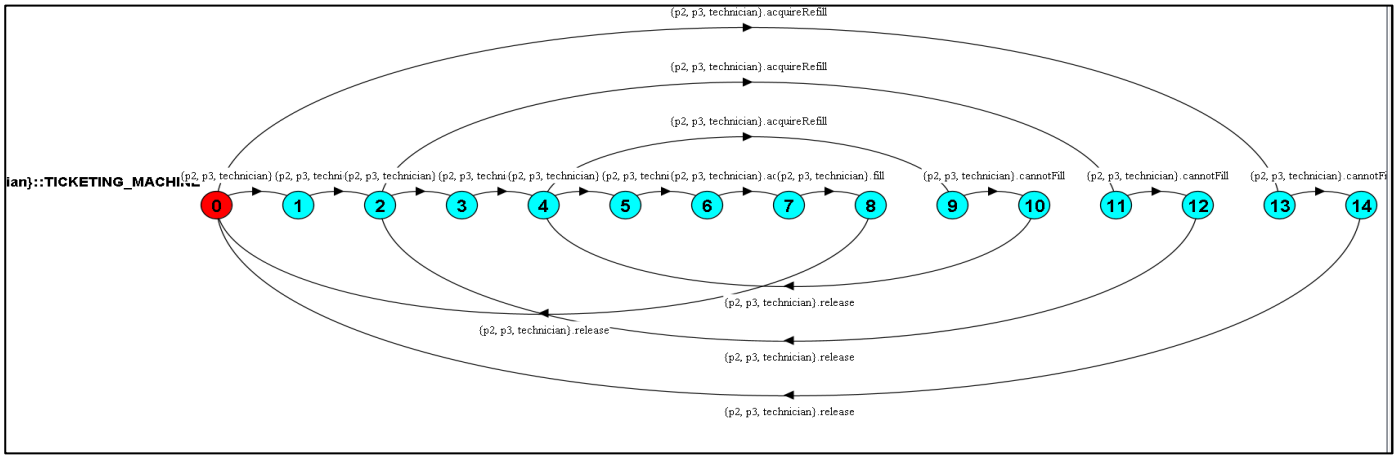


Figure 2- FSP process for TICKET_MACHINE

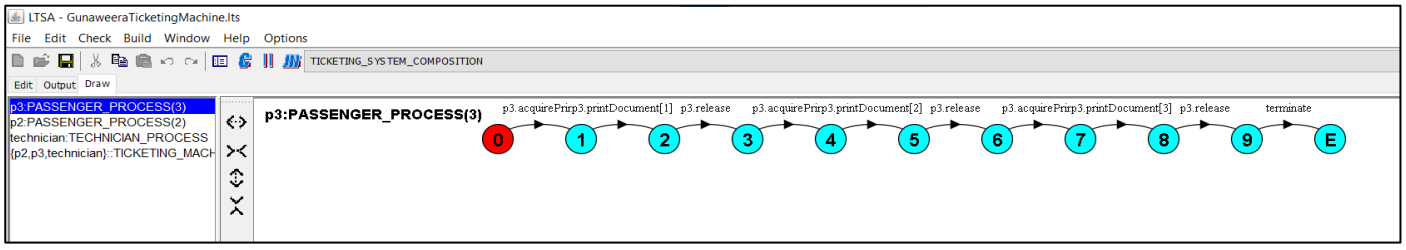


Figure 3- FSP process for PASSENGER3

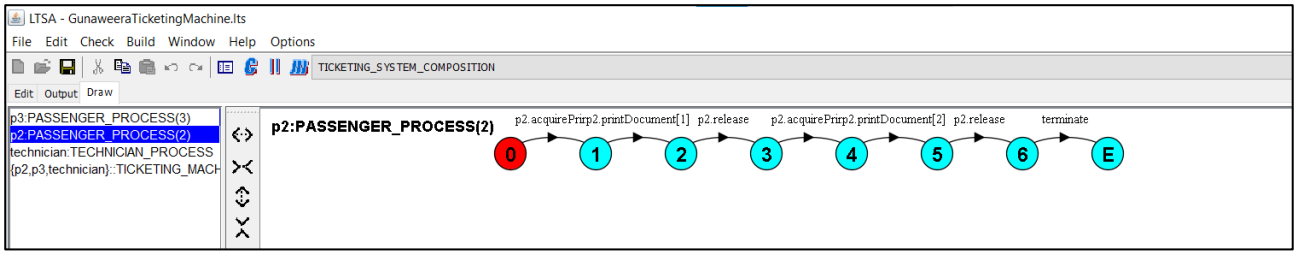


Figure 4- FSP process for PASSENGER2

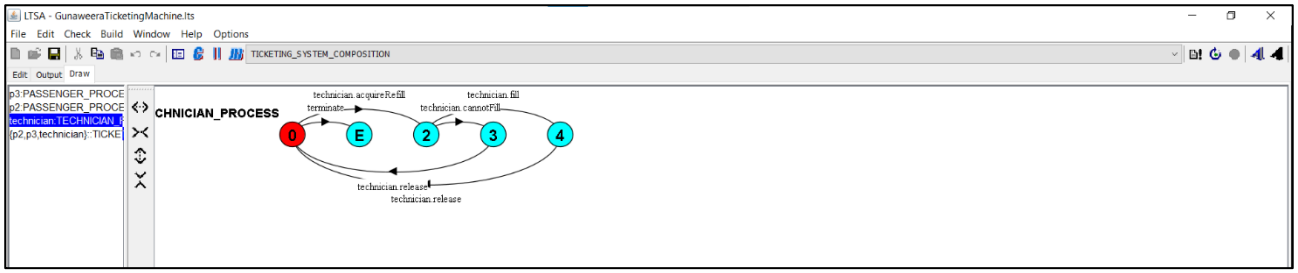


Figure 5- FSP process for TECHNICIAN

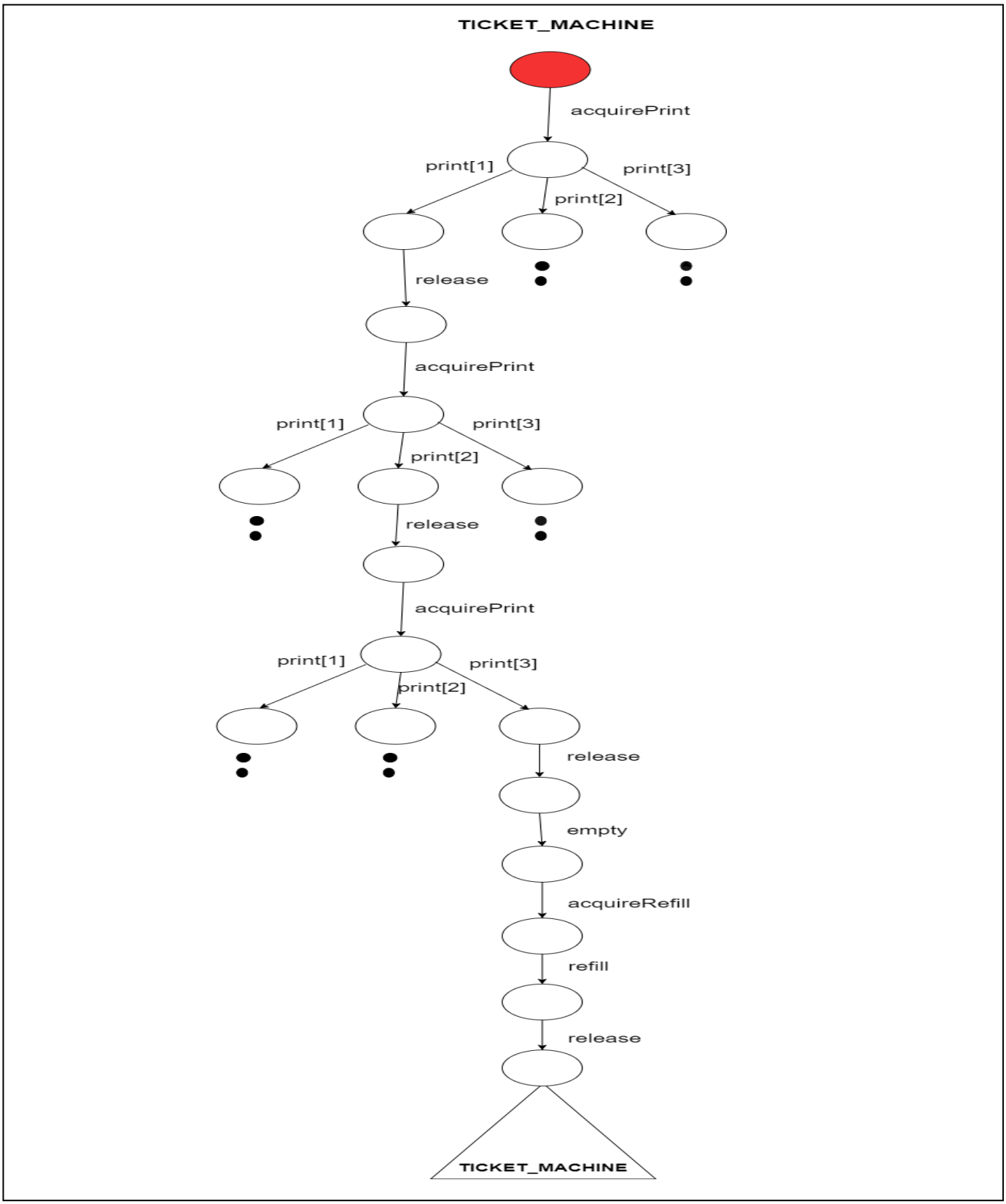
5. LTS States

A description of what each of the FSP process' states represents, i.e. is modelling. If there are a large number of states then you can group similar states together &/or only include the most important ones. For example, identify any states related to mutual exclusion (ME) & the associated critical section (CS), e.g. waiting to enter the CS state, in the CS state(s), left the CS state. (Add rows as necessary.)

State	Represents
Q0	The ticket machine containing 3 paper sheets and ready to be used. Waiting to enter the CS state.
Q1	The ticket machine is acquired by a process for printing the first ticket document. ME
Q2	The ticket machine has printed the first ticket document and ready to be released. Left the CS state.
Q3	After printing the first sheet, the ticket machine has been released and at this state ticket machine has 2 remaining sheets ready to print. Waiting to enter the CS state.
Q4	The ticket machine is acquired by a process for printing the second ticket document. ME
Q5	The ticket machine has printed the second ticket document and ready to be released. Left the CS state.
Q6	After printing the second sheet, the ticket machine has been released and at this state ticket machine has 1 remaining sheet ready to print. Waiting to enter the CS state.
Q7	The ticket machine is acquired by a process for printing the third ticket document. ME
Q8	The ticket machine has printed the third ticket document and ready to be released. Left the CS state.
Q9	After printing the last sheet, the ticket machine has been released and at this state printer has 0 sheets remaining and is ready to be refilled. Waiting to enter the CS state
Q10	The ticket machine has been acquired for refilling.
Q11	The ticket machine is waiting to be released, at this state ticket machine has 3 sheets of paper.

6. Trace Tree for FSP Process

The trace tree for the process. Use the conventions given in the lecture notes and add explanatory notes if necessary.



Above diagram represents the trace tree for a concurrent system denoted as TICKET_MACHINE. Each node within the tree denotes a distinct state or a specific point in the execution of the system, and the edges connecting the nodes illustrate the transitions between these states. Positioned at the bottom of the diagram is the TICKET_MACHINE process.

Above the TICKET_MACHINE, there is a box labeled "empty," signifying the initial state where the printer is devoid of paper. Adjacent to it is a box labeled "refillPrinter," indicating a process responsible for adding paper to the printer when it is empty.

Proceeding upward, three processes are depicted, each labeled as "acquire." These processes endeavor to acquire the printer. If the printer is available, the respective process gains the privilege to print; otherwise, it awaits the release of the printer by other processes.

Upon successfully acquiring the printer, each process transitions to one of three states: "print[1]," "print[2]," or "print[3]." These states correspond to different printing tasks that the process can execute. After completing the printing task, the process enters a state labeled "release," facilitating the release of the printer for subsequent acquisitions by other processes.

The "acquire" processes exhibit arrows looping back to themselves, indicating that once a process releases the printer, it can cyclically attempt to acquire it again. In essence, these loops signify repeated attempts to print until success.

It is crucial to note that the presented FSP (Finite State Process) trace tree diagram represents only one possible execution of the system. In reality, numerous alternative executions exist, contingent upon the scheduling of processes and the time required for printing tasks.