6SENG002W Concurrent Programming

FSP Process Analysis & Design Form

Name	Dulshan Ratnayake
Student ID	2017256/w1697707
Date	30/12/2020

1. FSP Process Attributes

Attribute	Value
Name	PRINTER
Description	This is the shared resource that is used by both students and technician. The process simulates the actions carried out by the printer
Alphabet	{print.acquire, refill.acquire,not.empty.acquire, not.empty, print[1], print[2], print[3], refill, release}
Number of States	15
Deadlocks (yes/no)	No
Deadlock Trace(s) (if applicable)	N/A

2. FSP Process Code

FSP Process:

```
//constants
const MAX_PAPER_LEVEL = 3
const MIN_PRINT_PAPER_LEVEL=1
const MIN_PAPER_LEVL=0
const MIN_PRINT_DOCUMENT = 1
const MIN_DOCUMENT=0
//ranges
range PRINT_Paper_Level =  MIN_PRINT_PAPER_LEVEL ..
MAX_PAPER_LEVEL
range PAPER_Level = MIN_PAPER_LEVL .. MAX_PAPER_LEVEL
set PRINT_Actions = { print[PRINT_Paper_Level],
print.acquire,refill.acquire,release,refill,not.empty.acquire,not.empty }
set PRINTER_Users = { tech, std1,std2}

//printer FINITE STATE MACHINE
PRINTER = PRINTER_FREE[MAX_PAPER_LEVEL] ,
PRINTER_FREE[papers_available : PAPER_Level] =
(when(papers_available>MIN_PAPER_LEVL)print.acquire ->
PRINT[papers_available]
|when(papers_available>MIN_PAPER_LEVL) not.empty.acquire->
EMPTY_PRINTER[papers_available]
|when(papers_available==MIN_PAPER_LEVL) refill.acquire->
REFILL_PRINTER[MAX_PAPER_LEVEL]) ,

EMPTY_PRINTER[papers_available : PRINT_Paper_Level]=(not.empty->
RELEASE_PRINTER[papers_available]),
REFILL_PRINTER[refill_papers : PAPER_Level]=(refill->
RELEASE_PRINTER[refill_papers]),
PRINT[papers_available : PRINT_Paper_Level] =(print[papers_available] ->
RELEASE_PRINTER[papers_available-1]),

RELEASE_PRINTER[papers : PAPER_Level]=(release->PRINTER_FREE[papers]).
```

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
print.acquire	A student acquiring the printer in order to print the documents	Synchronous
print[3]	A student printing his/her document by printing on the 3rd paper available in the printer.	Synchronous
print[2]	A student printing his/her document by printing on the 2nd paper available in the printer.	Synchronous
print[1]	A student printing his/her document by printing on the 1st paper available in the printer.	Synchronous
release	A student or a technician releasing the printer after using it	Synchronous
refill.acquire	Technician acquiring the printer in order refill the papers of the printer when all 3 papers of the printer has been used	Synchronous
refill	Technician refilling the printer with 3 papers	Synchronous
not.empty.acquire	Technician acquiring the printer to check if the printer is out of paper	Synchronous
not.empty	Technician finds that the printer has at least one page, thus the printer is not empty.	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.
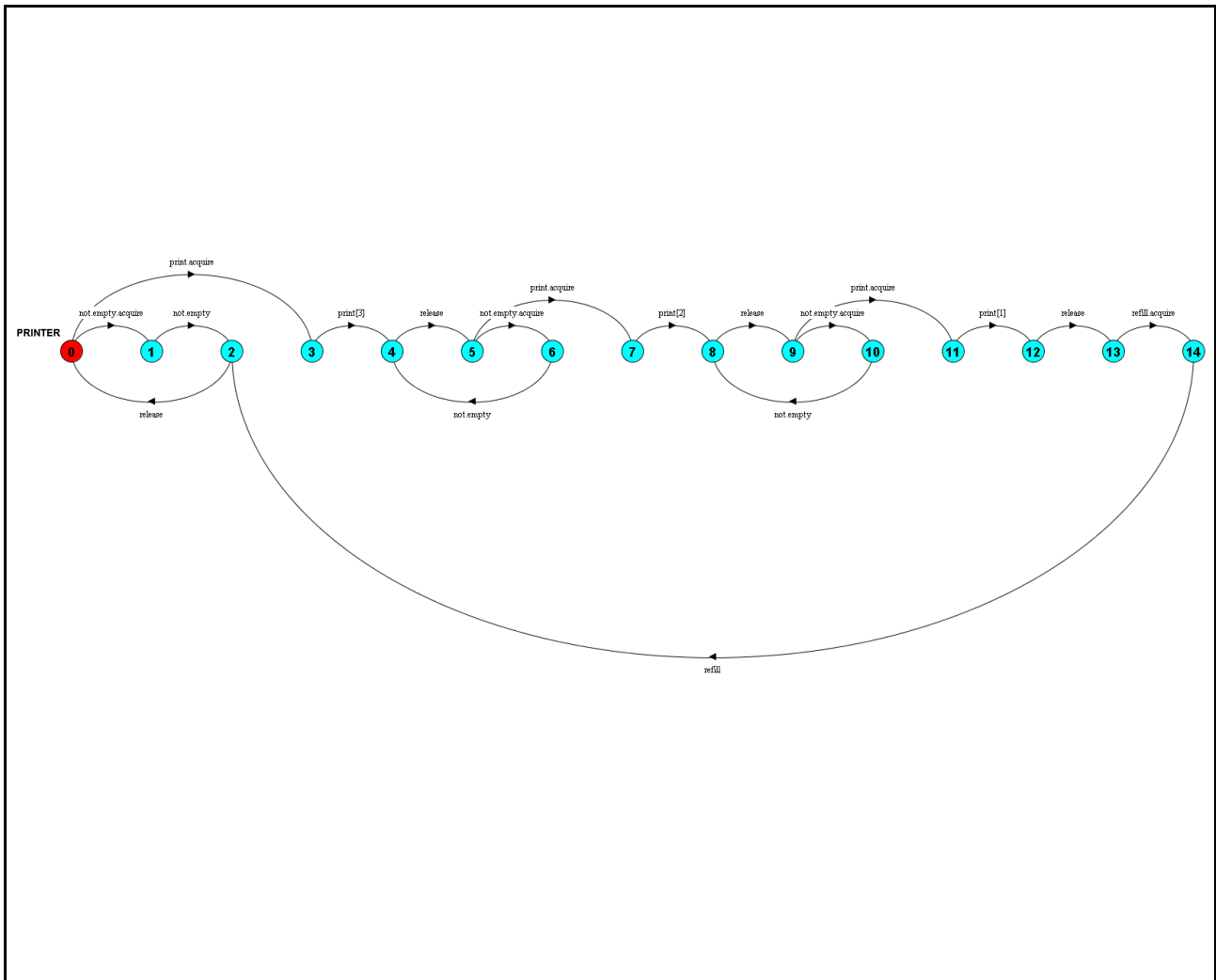
5. LTS States

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
0	A student can acquire the Printer to print or the technician can acquire to check whether its out of pages. If the technician has confirmed the paper tray is not empty or if the technician has refilled the printer with pages; both actions can lead to state 0 by following the release action.
1	Technician has acquired the printer to check whether its empty, i.e out of pages
2	The technician has checked and confirmed its not empty or the technician has refilled the paper tray with 3 papers.
3	A student has acquired the printer to print his document
4	The student has finished printing the document on the 3rd paper from the printer's paper tray. Technician rechecking the paper level and confirming the printer is not empty will cause to arrive at the same state as well.
5	The student/technician has released the printer after using it
6	Technician has acquired the printer to check whether its empty, i.e out of pages
7	A student has acquired the printer
8	The student has finished printing the document on the 2nd paper from the printer's paper tray. Technician rechecking the paper level and confirming the printer is not empty will cause to arrive at the same state as well.
9	The student/technician has released the printer
10	Technician has acquired the printer to check whether its empty, i.e out of pages
11	A student has acquired the printer
12	The student has finished printing the document on the 1st paper from the printer's paper tray.
13	The student has released the printer.
14	Technician has acquired the printer to refill the pages of the printer since all the papers on the paper tray have been used up. Invoking the refill action will cause to move to state 2 where the technician can release the printer.

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.
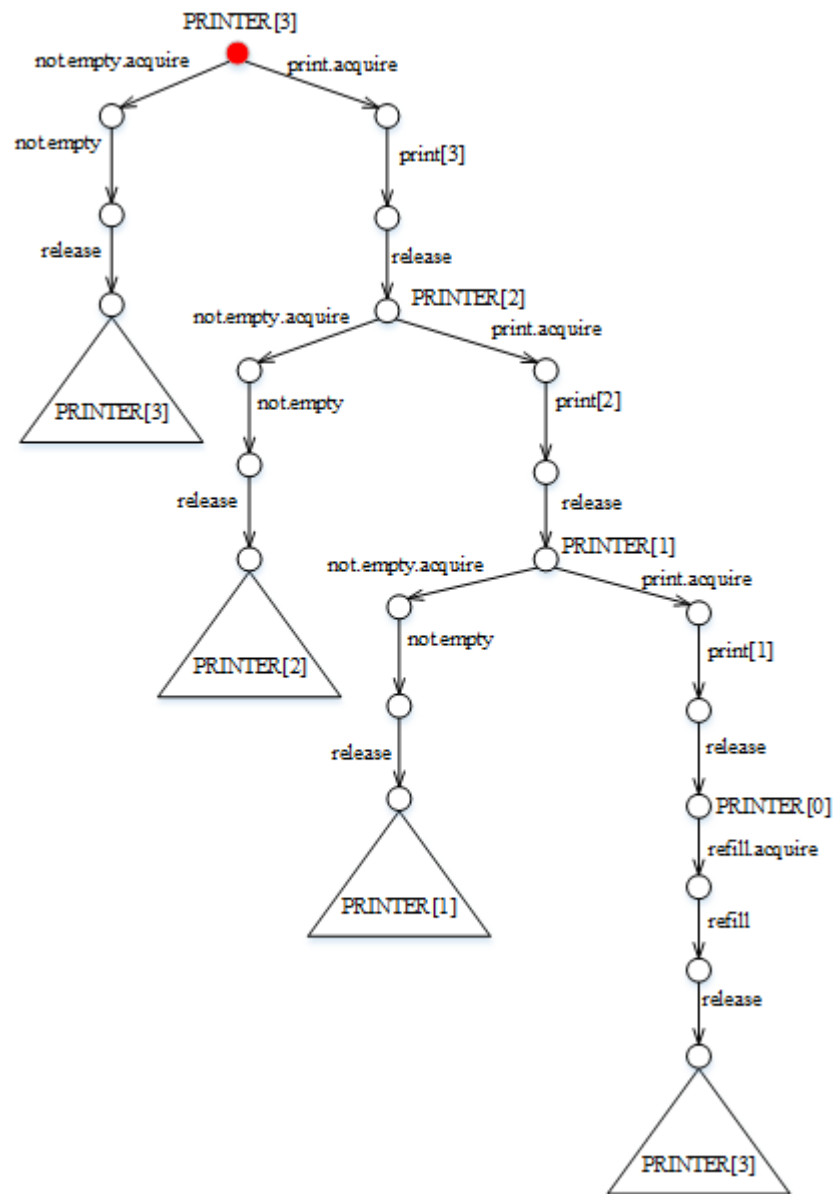
*Printer cannot have more than three papers on its paper tray.