

Report Midterm  
System Programming

Baran Hasan Bozduman  
171044036

May 3, 2021

# Problem Definition

Communicating between processes according to given story

## Design Decision

I initialized some named semaphores mutex(for the vaccinate part to use each time one vaccinator will use the buffer), empty(to control buffer limit), vac1 and vac2(I keep the vaccine types in two different semaphores for 1s and 2s), I get input from user and check them are they valid or not, then create two shared memory for citizens and citizen doses and after i create citizen processes I place them in the shared memory and terminate them im just using their pid ids then I create vaccinators and nurses nurses takes from file and post the vac1 and vac2 semaphores then until it comes to buffer limit. Vaccinators increases buffer sizes release by post empty and decreases the vaccines and since i keep my pid and doses in the shared memory i take and counter and start from last pid to begin and vaccinate them in the vaccinator. Also each vaccinator increases their counter to print how many doses they did and in the end i close semaphores and file for each processes so there is no leak then halt the program

```
[anonxx@Xmachina midterm]$ gcc -pthread -lrt program.c -Wall
[anonxx@Xmachina midterm]$ ./a.out -n 2 -c 3 -v 3 -b 10 -t 3 -l deneme
Nurse 0 (pid=312774) has brought vaccine 1: the clinic has 1 vaccine1 and 0 vaccine2.
Nurse 0 (pid=312774) has brought vaccine 2: the clinic has 1 vaccine1 and 1 vaccine2.
Nurse 0 (pid=312774) has brought vaccine 1: the clinic has 2 vaccine1 and 1 vaccine2.
Nurse 0 (pid=312774) has brought vaccine 1: the clinic has 3 vaccine1 and 1 vaccine2.
Nurse 0 (pid=312774) has brought vaccine 2: the clinic has 3 vaccine1 and 2 vaccine2.
Nurse 0 (pid=312774) has brought vaccine 2: the clinic has 3 vaccine1 and 3 vaccine2.
Nurse 0 (pid=312774) has brought vaccine 2: the clinic has 3 vaccine1 and 4 vaccine2.
Nurse 0 (pid=312774) has brought vaccine 2: the clinic has 3 vaccine1 and 5 vaccine2.
Nurse 0 (pid=312774) has brought vaccine 1: the clinic has 4 vaccine1 and 5 vaccine2.
Nurse 0 (pid=312774) has brought vaccine 1: the clinic has 5 vaccine1 and 5 vaccine2.
Vaccinator 0 (pid=312776) is inviting citizen pid=312773 to the clinic
Citizen 2 (pid=312773) is vaccinated for the 0th time: the clinic has 4 vaccine1 and 4 vaccine2.
Vaccinator 0 (pid=312776) is inviting citizen pid=312773 to the clinic
Nurse 0 (pid=312774) has brought vaccine 2: the clinic has 3 vaccine1 and 4 vaccine2.
Nurse 0 (pid=312774) has brought vaccine 2: the clinic has 3 vaccine1 and 5 vaccine2.
Citizen 2 (pid=312773) is vaccinated for the 1th time: the clinic has 3 vaccine1 and 5 vaccine2.
Vaccinator 0 (pid=312776) is inviting citizen pid=312773 to the clinic
Nurse 0 (pid=312774) has brought vaccine 1: the clinic has 3 vaccine1 and 4 vaccine2.
Citizen 2 (pid=312773) is vaccinated for the 2th time: the clinic has 3 vaccine1 and 4 vaccine2.
Nurse 0 (pid=312774) has brought vaccine 1: the clinic has 4 vaccine1 and 4 vaccine2.
Nurse 0 (pid=312774) has brought vaccine 1: the clinic has 5 vaccine1 and 4 vaccine2.
Vaccinator 0 (pid=312776) is inviting citizen pid=312772 to the clinic
Nurse 0 (pid=312774) has brought vaccine 2: the clinic has 4 vaccine1 and 4 vaccine2.
Citizen 1 (pid=312772) is vaccinated for the 0th time: the clinic has 4 vaccine1 and 4 vaccine2.
Vaccinator 0 (pid=312776) is inviting citizen pid=312772 to the clinic
Nurse 0 (pid=312774) has brought vaccine 1: the clinic has 4 vaccine1 and 3 vaccine2.
Citizen 1 (pid=312772) is vaccinated for the 1th time: the clinic has 4 vaccine1 and 3 vaccine2.
Nurse 0 (pid=312774) has brought vaccine 2: the clinic has 4 vaccine1 and 4 vaccine2.
Vaccinator 0 (pid=312776) is inviting citizen pid=312772 to the clinic
Citizen 1 (pid=312772) is vaccinated for the 2th time: the clinic has 3 vaccine1 and 3 vaccine2.
Vaccinator 0 (pid=312776) is inviting citizen pid=312771 to the clinic
Citizen 0 (pid=312771) is vaccinated for the 0th time: the clinic has 2 vaccine1 and 2 vaccine2.
Vaccinator 2 (pid=312778) is inviting citizen pid=312771 to the clinic
Citizen 0 (pid=312771) is vaccinated for the 1th time: the clinic has 1 vaccine1 and 1 vaccine2.
Vaccinator 2 (pid=312778) is inviting citizen pid=312771 to the clinic
Citizen 0 (pid=312771) is vaccinated for the 2th time: the clinic has 0 vaccine1 and 0 vaccine2.
All citizens have been vaccinated.
Vaccinator 2 (pid=312778) vaccinated 2 doses. Vaccinator 1 (pid=312777) vaccinated 0 doses. Vaccinator 0 (pid=312776) vaccinated 7 doses. The clinic is now closed. Stay healthy.
[anonxx@Xmachina midterm]$
```

## The Requirements I have achieved

For the bonus part as you see above I can vaccinate them in pid order but first it makes all doses for older citizen then it goes forward to next process ids I tried the rules below and they worked properly on my computer

1. No compilation error
2. No compilation warning
3. *Makefile*
4. *Report in LaTeX format*
5. Printed usage information in case of missing or invalid argument
6. Program did not crashed

7. No memory leaks
8. No zombie process
9. No deadlock
10. No poor synchronization
11. Submitted on time
12. Informed user in case of system errors
13. If all processes get created, and at least one citizen is vaccinated at least once, yes.

## **The Requirements I have failed**

I could not handled CTRL+C termination signal