# CSE344 Midterm

Processes and semaphores

A Producer Consumer Solution

Hüsnü AKÇAK
161044112

# Program Setup

## Shared Memory Segment

```c
struct ShmStatic* shmStatic; // conditional vars and unnamed semaphores
char *shmBuff;  // vaccine buffer
struct Citizen* shmCitizens; // to invite citizens to the clinic
int *shmChildPids; // to be able to send signal between any processes
```

## Structs

```c
struct ShmStatic {
    int buffSize,           // current number of vaccine
        buffCapacity,       // vaccine buffer capacity
        totalNumOfCitizens,
        leftCitizens,       // remaining citizen
        terminate, // a sign for others to terminate
        leftNurses,// to know which nurse prints their termination message
        vaccinationSession, // t*c
        sigIntArrived;      // to tell the other proc that SIGINT is received

    sem_t semEmpty, // initial value is buffer capacity
        semCitizenAvailable, // num of citizens waiting to be invited
        shmLock,    // lock all shared memory segment
        semVacc1,   // to notify and wait Vaccine '1'
        semVacc2,   // to notify and wait Vaccine '2'
        semVaccAvailable;  // at least a pair of '1' and '2' is available
};

struct Citizen{
    int pid;
    char gone,      // citizen leaved
         inClinic;  // currently busy in clinic
};
```

## Intoduction

At the begining, parent process prepares shared memory segment, opens the input file, establishes signal handlers for SIGUSR1 and SIGINT. Then fork children in this order, citizens, nurses, vaccinators, pushers. While forking citizens, fills "shmCitizens" array as citizen pid's are going to be ascending order. During the execution of the program parent process acts as "PusherV2" and at the end destroys semaphores, performs munmap and unlink for shared memory segment, wait for children then terminate.

## File reading

Nurses use the same fd to read input file, it is assumed there are enough '1' and '2' pairs of shot. While reading the file irrelavent characters(other then '1' and '2') are ignored.

## Nurse

The syscronization among nurses on input file is solved with flock(), each nurse take the "LOCK EX" and when she finishes her transportation gives the lock with "LOCK UN". When a nurse puts '1' into the buffer she also posts "semVacc1", if ’2’ is delivered "semVacc2" is posted. When EOF is encountered they terminated one by one, the last nurse print nurses termination message then she also gone.

## Pusher V1

Waits for "semVacc1" in the number of expected time. When vaccine ’1’ is received it controls existence of vaccine ’2’ in the buffer if it exist post "semVaccAvailable" this means a complete shot is ready.

## Pusher V2

Pusher V2 follows the same steps with Pusher V1 in reverse.

## Vaccinator

Each vaccinator knows how many dose are going to be applied in total and follows the current remaining dose from shared memory segment. Until this count reaches zero they wait for vaccine ’1’ and vaccine ’2’ pairs by waiting "semVaccAvailable". When there is at least one vaccine pair the vaccinator removes them from the buffer (while having "shmLock"). Waits for "semCitizenAvailable" and invites the oldest one of them, vaccinate the citizen and post "semEmpty" for buffer two times. Then goes back start of the loop, and continues until total shots are applied.

## How to choose and invite the oldest citizen?

The array, "shmCitizens" keeps all citizens in ascending order by their process ids. Also, it is observable through this array whether the citizen leaves the clinic or is being vaccinated right now in the clinic. In the light of this this information, the oldest available citizen is invited by sending SIGUSR1 to him.

## SIGUSR1 setup

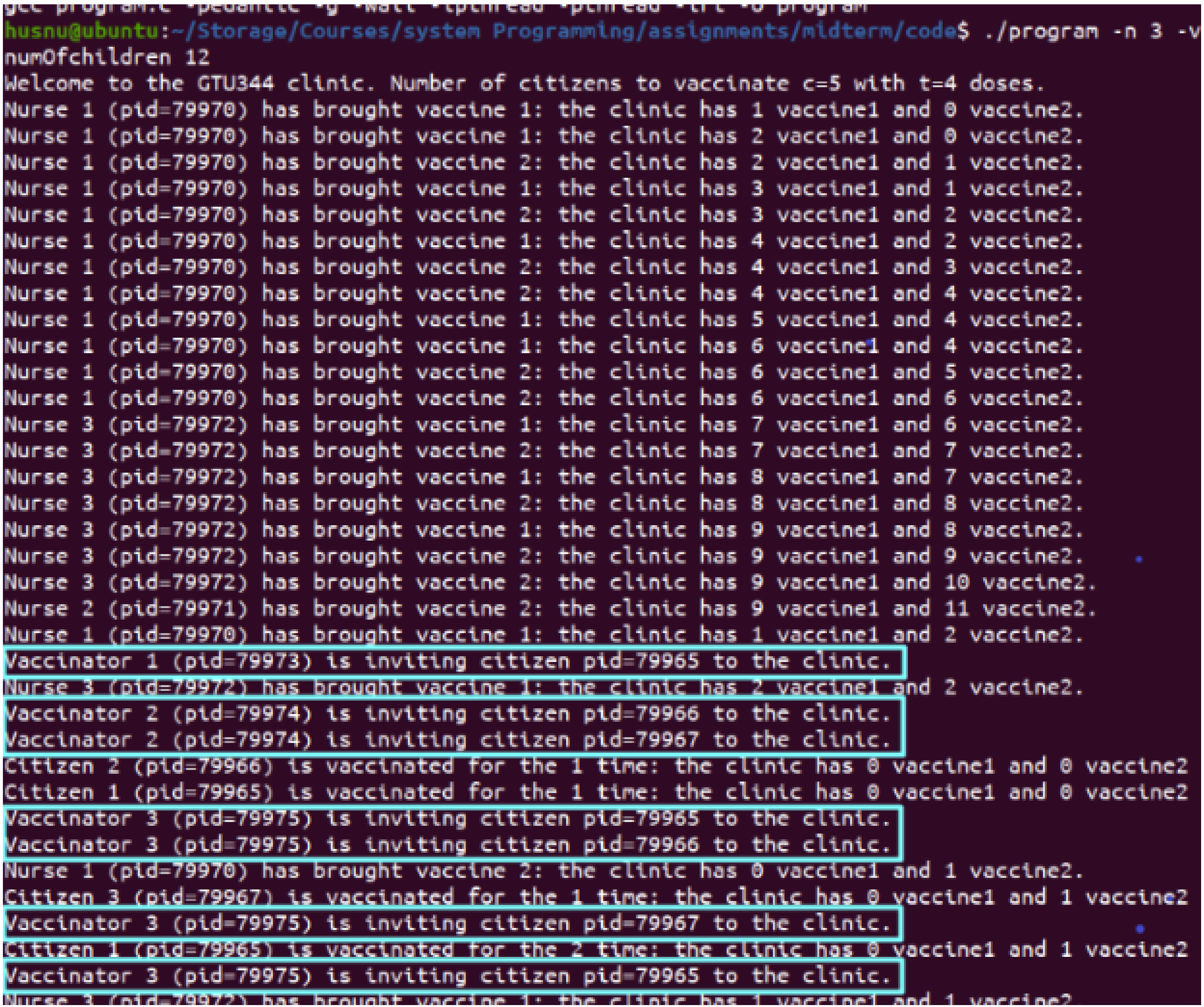
To be able to communicate with citizens through SIGUSR1 first sigprocmask is used to block the signal, after wards sigsuspend is used in citizen code to wait an invitation from a vaccinator.

## Citizen

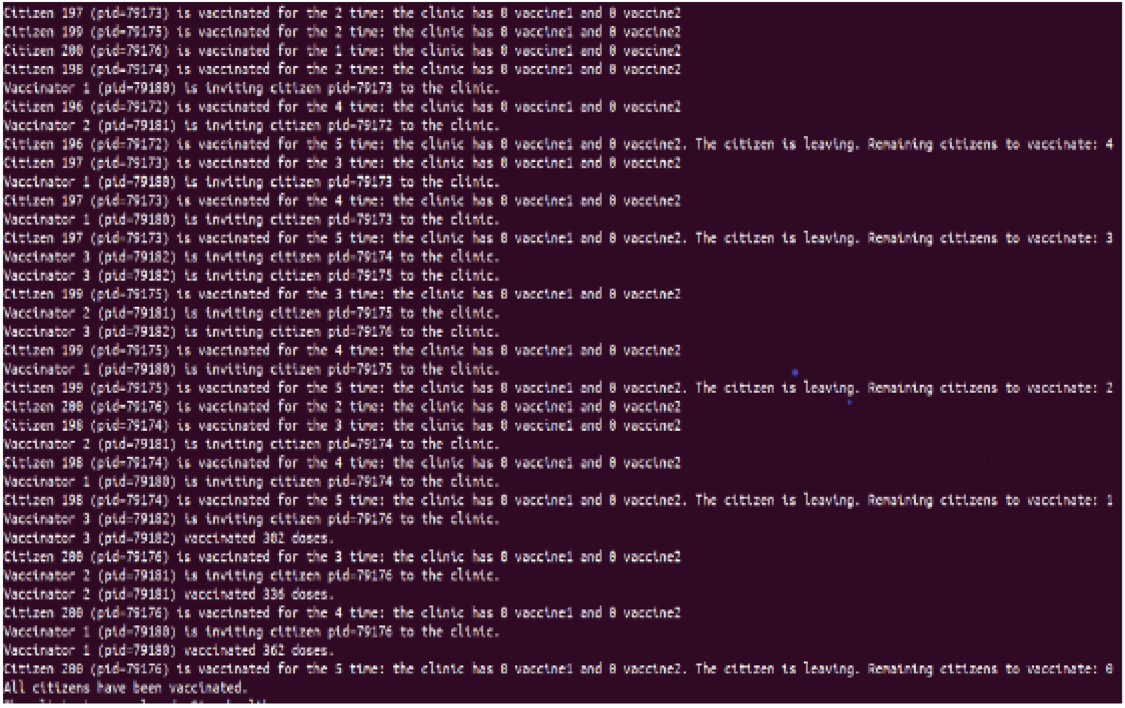
Every citizen has a while loop which is executed t times as soon as the loop starts, SIGUSR1 is being waited by sigsuspend. When the signal is arrived the citizen takes her dose, if she gets vaccinated "t" times she leaved the clinic, otherwise she post semCitizenAvailable then waits to be invited again.

In general it is challenging to deal with signals, but in this scenario the signal is send by the vaccinator while he have a lock, it is guaranteed that only one signal is arriving to the same citizen at a time.

## Screen Shots



```
gcc program.c -pedantic -g -Wall -lpthread -pthread -lrt -o program
husnu@ubuntu:~/Storage/Courses/system Programming/assignments/midterm/code$ ./program -n 3 -v
numOfchildren 12
Welcome to the GTU344 clinic. Number of citizens to vaccinate c=5 with t=4 doses.
Nurse 1 (pid=79970) has brought vaccine 1: the clinic has 1 vaccine1 and 0 vaccine2.
Nurse 1 (pid=79970) has brought vaccine 1: the clinic has 2 vaccine1 and 0 vaccine2.
Nurse 1 (pid=79970) has brought vaccine 2: the clinic has 2 vaccine1 and 1 vaccine2.
Nurse 1 (pid=79970) has brought vaccine 1: the clinic has 3 vaccine1 and 1 vaccine2.
Nurse 1 (pid=79970) has brought vaccine 2: the clinic has 3 vaccine1 and 2 vaccine2.
Nurse 1 (pid=79970) has brought vaccine 1: the clinic has 4 vaccine1 and 2 vaccine2.
Nurse 1 (pid=79970) has brought vaccine 2: the clinic has 4 vaccine1 and 3 vaccine2.
Nurse 1 (pid=79970) has brought vaccine 2: the clinic has 4 vaccine1 and 4 vaccine2.
Nurse 1 (pid=79970) has brought vaccine 1: the clinic has 5 vaccine1 and 4 vaccine2.
Nurse 1 (pid=79970) has brought vaccine 1: the clinic has 6 vaccine1 and 4 vaccine2.
Nurse 1 (pid=79970) has brought vaccine 2: the clinic has 6 vaccine1 and 5 vaccine2.
Nurse 1 (pid=79970) has brought vaccine 2: the clinic has 6 vaccine1 and 6 vaccine2.
Nurse 3 (pid=79972) has brought vaccine 1: the clinic has 7 vaccine1 and 6 vaccine2.
Nurse 3 (pid=79972) has brought vaccine 2: the clinic has 7 vaccine1 and 7 vaccine2.
Nurse 3 (pid=79972) has brought vaccine 1: the clinic has 8 vaccine1 and 7 vaccine2.
Nurse 3 (pid=79972) has brought vaccine 2: the clinic has 8 vaccine1 and 8 vaccine2.
Nurse 3 (pid=79972) has brought vaccine 1: the clinic has 9 vaccine1 and 8 vaccine2.
Nurse 3 (pid=79972) has brought vaccine 2: the clinic has 9 vaccine1 and 9 vaccine2.
Nurse 3 (pid=79972) has brought vaccine 2: the clinic has 9 vaccine1 and 10 vaccine2.
Nurse 2 (pid=79971) has brought vaccine 2: the clinic has 9 vaccine1 and 11 vaccine2.
Nurse 1 (pid=79970) has brought vaccine 1: the clinic has 1 vaccine1 and 2 vaccine2.
Vaccinator 1 (pid=79973) is inviting citizen pid=79965 to the clinic.
Nurse 3 (pid=79972) has brought vaccine 1: the clinic has 2 vaccine1 and 2 vaccine2.
Vaccinator 2 (pid=79974) is inviting citizen pid=79966 to the clinic.
Vaccinator 2 (pid=79974) is inviting citizen pid=79967 to the clinic.
Citizen 2 (pid=79966) is vaccinated for the 1 time: the clinic has 0 vaccine1 and 0 vaccine2
Citizen 1 (pid=79965) is vaccinated for the 1 time: the clinic has 0 vaccine1 and 0 vaccine2
Vaccinator 3 (pid=79975) is inviting citizen pid=79965 to the clinic.
Vaccinator 3 (pid=79975) is inviting citizen pid=79966 to the clinic.
Nurse 1 (pid=79970) has brought vaccine 2: the clinic has 0 vaccine1 and 1 vaccine2.
Citizen 3 (pid=79967) is vaccinated for the 1 time: the clinic has 0 vaccine1 and 1 vaccine2
Vaccinator 3 (pid=79975) is inviting citizen pid=79967 to the clinic.
Citizen 1 (pid=79965) is vaccinated for the 2 time: the clinic has 0 vaccine1 and 1 vaccine2
Vaccinator 3 (pid=79975) is inviting citizen pid=79965 to the clinic.
Nurse 3 (pid=79972) has brought vaccine 1: the clinic has 1 vaccine1 and 1 vaccine2
```



```
Citizen 197 (pid=79173) is vaccinated for the 2 time: the clinic has 0 vaccine1 and 0 vaccine2
Citizen 199 (pid=79175) is vaccinated for the 2 time: the clinic has 0 vaccine1 and 0 vaccine2
Citizen 200 (pid=79176) is vaccinated for the 1 time: the clinic has 0 vaccine1 and 0 vaccine2
Citizen 198 (pid=79174) is vaccinated for the 2 time: the clinic has 0 vaccine1 and 0 vaccine2
Vaccinator 1 (pid=79180) is inviting citizen pid=79173 to the clinic.
Citizen 196 (pid=79172) is vaccinated for the 4 time: the clinic has 0 vaccine1 and 0 vaccine2
Vaccinator 2 (pid=79181) is inviting citizen pid=79172 to the clinic.
Citizen 196 (pid=79172) is vaccinated for the 5 time: the clinic has 0 vaccine1 and 0 vaccine2. The citizen is leaving. Remaining citizens to vaccinate: 4
Citizen 197 (pid=79173) is vaccinated for the 3 time: the clinic has 0 vaccine1 and 0 vaccine2
Vaccinator 1 (pid=79180) is inviting citizen pid=79173 to the clinic.
Citizen 197 (pid=79173) is vaccinated for the 4 time: the clinic has 0 vaccine1 and 0 vaccine2
Vaccinator 1 (pid=79180) is inviting citizen pid=79173 to the clinic.
Citizen 197 (pid=79173) is vaccinated for the 5 time: the clinic has 0 vaccine1 and 0 vaccine2. The citizen is leaving. Remaining citizens to vaccinate: 3
Vaccinator 3 (pid=79182) is inviting citizen pid=79174 to the clinic.
Vaccinator 3 (pid=79182) is inviting citizen pid=79175 to the clinic.
Citizen 199 (pid=79175) is vaccinated for the 3 time: the clinic has 0 vaccine1 and 0 vaccine2
Vaccinator 2 (pid=79181) is inviting citizen pid=79175 to the clinic.
Vaccinator 3 (pid=79182) is inviting citizen pid=79176 to the clinic.
Citizen 199 (pid=79175) is vaccinated for the 4 time: the clinic has 0 vaccine1 and 0 vaccine2
Vaccinator 1 (pid=79180) is inviting citizen pid=79175 to the clinic.
Citizen 199 (pid=79175) is vaccinated for the 5 time: the clinic has 0 vaccine1 and 0 vaccine2. The citizen is leaving. Remaining citizens to vaccinate: 2
Citizen 200 (pid=79176) is vaccinated for the 2 time: the clinic has 0 vaccine1 and 0 vaccine2
Citizen 198 (pid=79174) is vaccinated for the 3 time: the clinic has 0 vaccine1 and 0 vaccine2
Vaccinator 2 (pid=79181) is inviting citizen pid=79174 to the clinic.
Citizen 198 (pid=79174) is vaccinated for the 4 time: the clinic has 0 vaccine1 and 0 vaccine2
Vaccinator 1 (pid=79180) is inviting citizen pid=79174 to the clinic.
Citizen 198 (pid=79174) is vaccinated for the 5 time: the clinic has 0 vaccine1 and 0 vaccine2. The citizen is leaving. Remaining citizens to vaccinate: 1
Vaccinator 3 (pid=79182) is inviting citizen pid=79176 to the clinic.
Vaccinator 3 (pid=79182) vaccinated 302 doses.
Citizen 200 (pid=79176) is vaccinated for the 3 time: the clinic has 0 vaccine1 and 0 vaccine2
Vaccinator 2 (pid=79181) is inviting citizen pid=79176 to the clinic.
Vaccinator 2 (pid=79181) vaccinated 336 doses.
Citizen 200 (pid=79176) is vaccinated for the 4 time: the clinic has 0 vaccine1 and 0 vaccine2
Vaccinator 1 (pid=79180) is inviting citizen pid=79176 to the clinic.
Vaccinator 1 (pid=79180) vaccinated 362 doses.
Citizen 200 (pid=79176) is vaccinated for the 5 time: the clinic has 0 vaccine1 and 0 vaccine2. The citizen is leaving. Remaining citizens to vaccinate: 0
All citizens have been vaccinated.
```