Gebze Technical University
Computer Engineering

System Programming
CSE344 – 2021

MIDTERM-REPORT

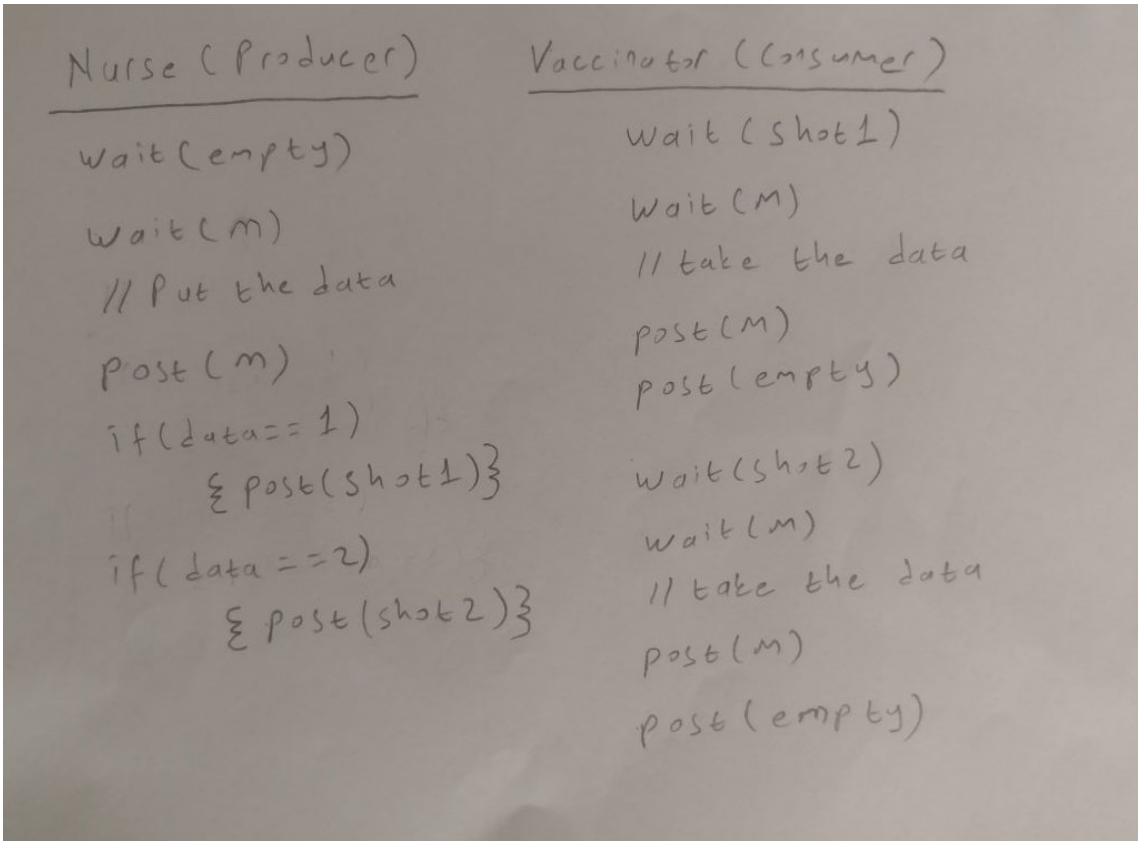
Yusuf Abdullah ARSLANALP
151044046

0.1 How I Solved This Problem

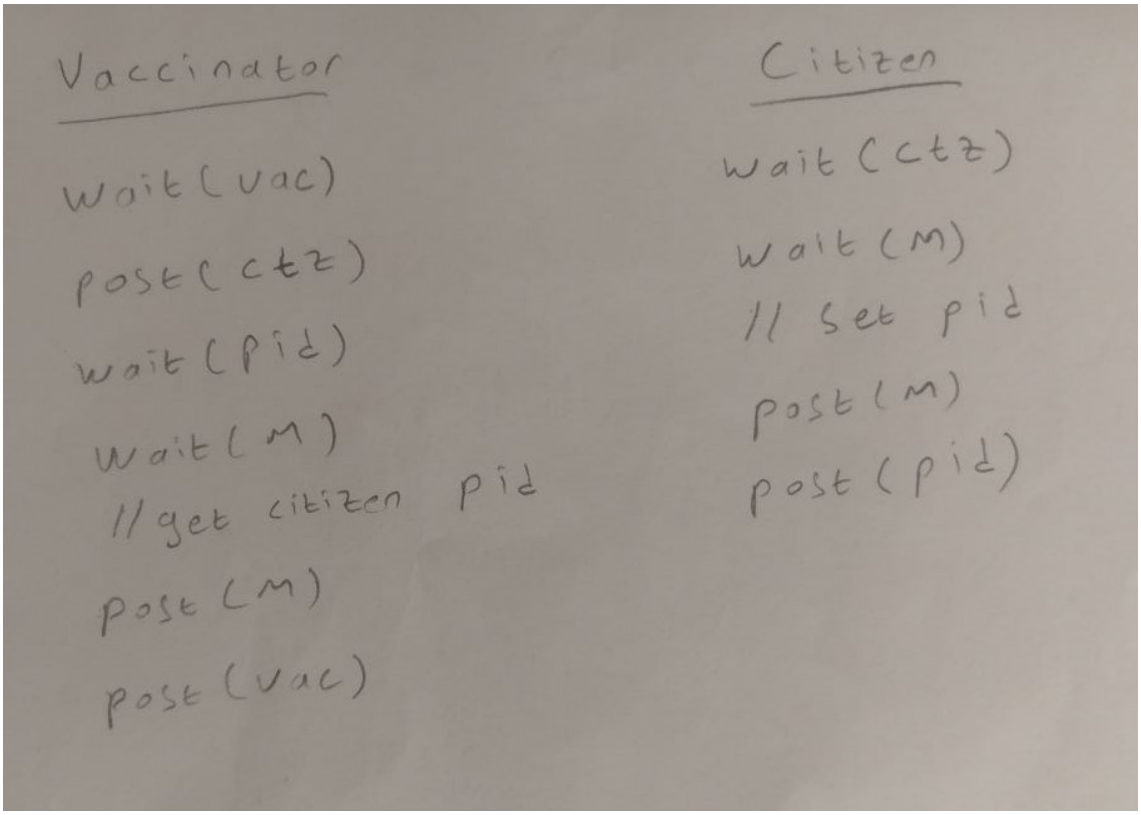
I used 7 semaphore to solve the problem. Semaphores are as follows:

```
27  sem_t * sem_empty;      //initially buffer size
28  sem_t * sem_csection;   //initially 1
29  sem_t * sem_shot1;      //initially 0
30  sem_t * sem_shot2;      //initially 0
31  sem_t * sem_vac;        //initially 1
32  sem_t * sem_client;     //initially 0
33  sem_t * sem_pid;        //initially 0
```

The pseudocode for nurse-vaccinator synchronization is as follows:



The pseudocode for nurse-vaccinator synchronization is as follows:



0.2 To solve this problem in addition to semaphores I used:

- named fifo
- Signal
- Named shared memory

0.3 Which requirements I achieved

- I Protect the buffer against underflow and overflow
- No race condition for the buffer
- Input file is read sequentially.
- When CTRL-C pressed all processes terminated. And all resources are given back to the system.
- I used only 7 semaphore
- No warning with -Wall flag

- If the required command line arguments are missing/invalid, The program prints usage information and exit.
- The report prepeared via latex. (latex folder is in the homework)
- No zombie processes
- No busy waiting. No sleep.
- The make file only compiles the program.
- No memory leaks.

```
==16479==
==16479== HEAP SUMMARY:
==16479==     in use at exit: 0 bytes in 0 blocks
==16479==   total heap usage: 14 allocs, 14 frees, 441 bytes allocated
==16479==
```