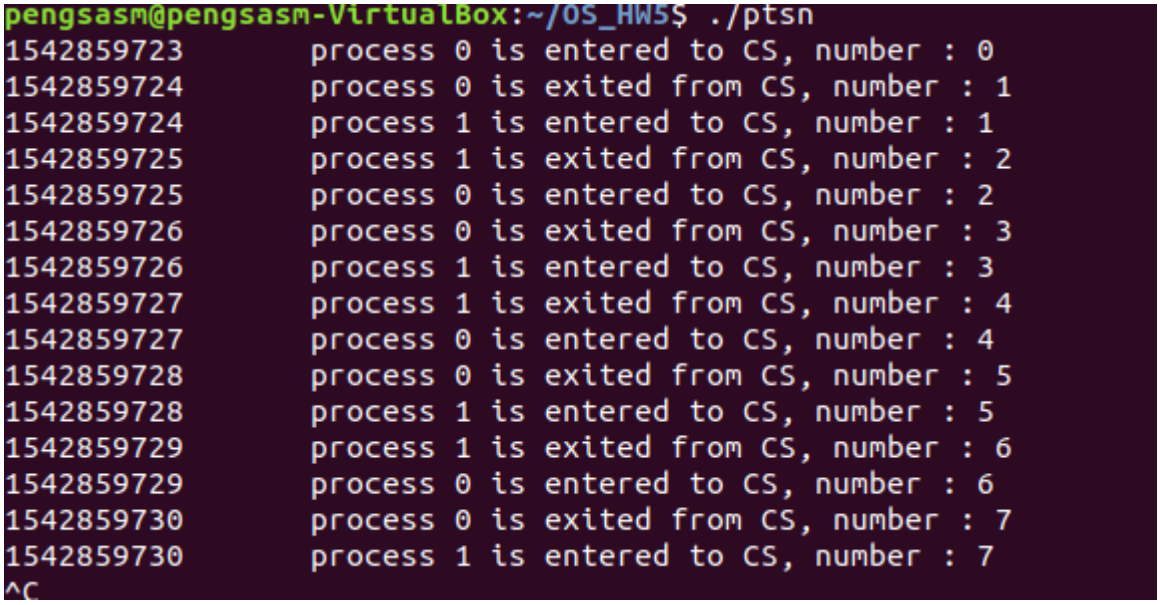
4. Implement the Peterson's algorithm in lecture note 6-12 using semaphore. Run the program, and show if it works correctly by using a time chart. The chart must show the events as they occur. (25)

```c
#include <stdio.h>
#include <semaphore.h>
#include <time.h>
#include <sys/types.h>
#include <sys/mman.h>
#include <unistd.h>
#include <stdlib.h>

static int* turn;
static int* shrd_num;
static sem_t* mutex;


int main(){
        turn = mmap(0, sizeof(*turn), PROT_READ|PROT_WRITE, MAP_SHARED | MAP_ANONYMOUS, -1, 0);
        shrd_num = mmap(0, sizeof(*shrd_num), PROT_READ|PROT_WRITE, MAP_SHARED | MAP_ANONYMOUS, -1, 0);
        mutex = mmap(0, sizeof(*mutex), PROT_READ|PROT_WRITE, MAP_SHARED | MAP_ANONYMOUS, -1, 0);

        sem_init(mutex,1,1);
        *shrd_num = 0;
        *turn = 0;

        pid_t pid = fork();

        if(pid < 0)
                exit(1);

       if(pid > 0)
       {
               do{
                       *turn = 1;
                       while(*turn == 1);
                       sem_wait(mutex);
                       printf("%ld\t process 0 is entered to CS, number : %d\n",
                                       time(NULL), *shrd_num);

                       (*shrd_num)++;
                       sleep(1);

                       sem_post(mutex);
                       printf("%ld\t process 0 is exited from CS, number : %d\n",
                                       time(NULL), *shrd_num);
               }while(1);

       }

       else
       {
               do{
                       *turn = 0;
                       while(*turn == 0);
                       sem_wait(mutex);
                       printf("%ld\t process 1 is entered to CS, number : %d\n",
                                       time(NULL), *shrd_num);

                       (*shrd_num)++;
                       sleep(1);

                       sem_post(mutex);
                       printf("%ld\t process 1 is exited from CS, number : %d\n",
                                  time(NULL), *shrd_num);
               }while(1);


       }

       return 0;

}
```

- code


pengsasm@pengsasm-VirtualBox:~/OS_HW5$ ./ptsn
1542859723 process 0 is entered to CS, number : 0
1542859724 process 0 is exited from CS, number : 1
1542859724 process 1 is entered to CS, number : 1
1542859725 process 1 is exited from CS, number : 2
1542859725 process 0 is entered to CS, number : 2
1542859726 process 0 is exited from CS, number : 3
1542859726 process 1 is entered to CS, number : 3
1542859727 process 1 is exited from CS, number : 4
1542859727 process 0 is entered to CS, number : 4
1542859728 process 0 is exited from CS, number : 5
1542859728 process 1 is entered to CS, number : 5
1542859729 process 1 is exited from CS, number : 6
1542859729 process 0 is entered to CS, number : 6
1542859730 process 0 is exited from CS, number : 7
1542859730 process 1 is entered to CS, number : 7
^C

- result