Name : Bramha Nimbalkar

Roll no: 7

Srn : 202100381

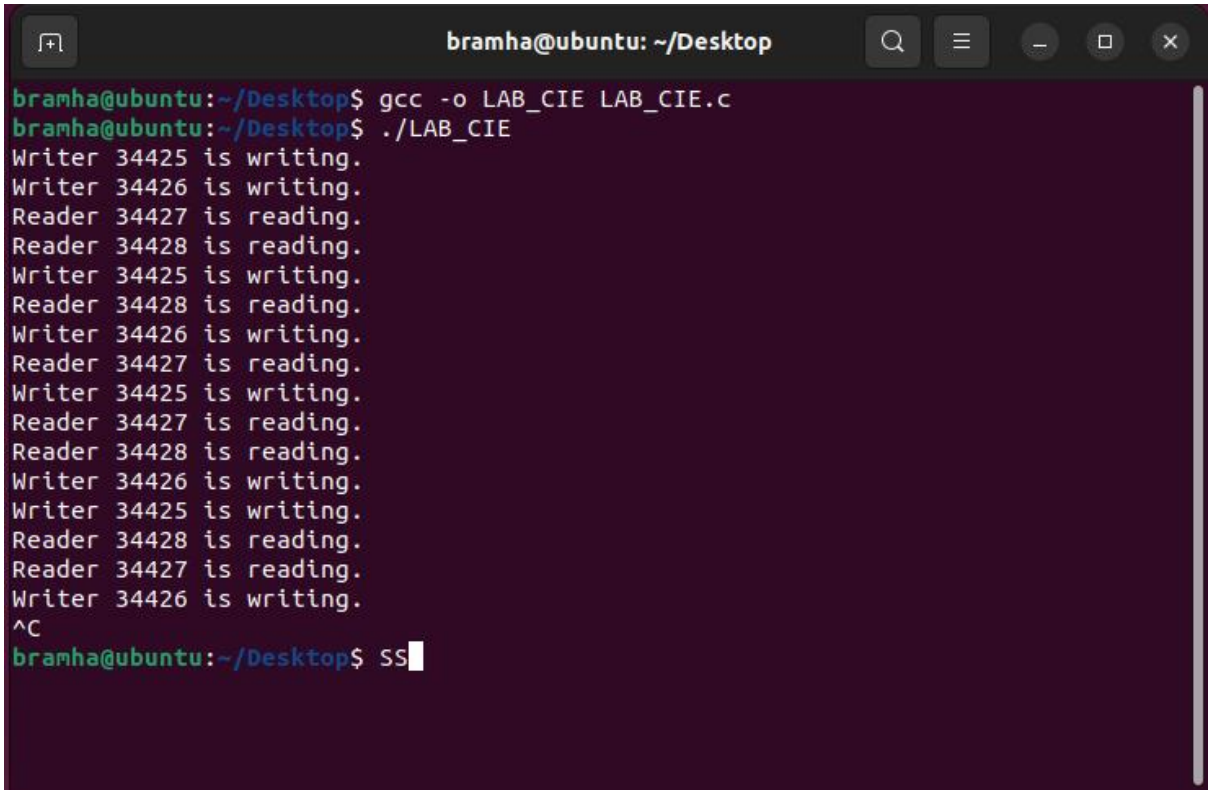
LAB CIE

Implement Reader Writer problem using semaphore where u have to use fork() system call for creating child process.

```c
#include <stdio.h>
#include <stdlib.h>
#include <pthread.h>
#include <semaphore.h>
#include <unistd.h>
#include <sys/types.h>
#include <sys/wait.h>


#define NUM_READERS 2
#define NUM_WRITERS 2

sem_t mutex;
sem_t db;
int rCount = 0;

void writer() {
    while (1) {
        sem_wait(&db);
        printf("Writer %d is writing.\n", getpid());
        sleep(2);
        sem_post(&db);
    }
}

void reader() {
    while (1) {
        sem_wait(&mutex);
        rCount++;
        if (rCount == 1) {
            sem_wait(&db);
        }
```

```
        sem_post(&mutex);

        printf("Reader %d is reading.\n", getpid());
        sleep(2);

        sem_wait(&mutex);
        rCount--;
        if (rCount == 0) {
            sem_post(&db);
        }
        sem_post(&mutex);
    }
}

int main() {
    sem_init(&mutex, 0, 1);
    sem_init(&db, 0, 1);

    for (int i = 0; i < NUM_WRITERS; i++) {
        if (fork() == 0) {
            writer();
            exit(0);
        }
    }

    for (int i = 0; i < NUM_READERS; i++) {
        if (fork() == 0) {
            reader();
            exit(0);
        }
    }

    while (wait(NULL) > 0);
    sem_destroy(&mutex);
    sem_destroy(&db);

    return 0;
}
```

Output :


bramha@ubuntu: ~/Desktop
bramha@ubuntu:~/Desktop$ gcc -o LAB_CIE LAB_CIE.c
bramha@ubuntu:~/Desktop$ ./LAB_CIE
Writer 34425 is writing.
Writer 34426 is writing.
Reader 34427 is reading.
Reader 34428 is reading.
Writer 34425 is writing.
Reader 34428 is reading.
Writer 34426 is writing.
Reader 34427 is reading.
Writer 34425 is writing.
Reader 34427 is reading.
Reader 34428 is reading.
Writer 34426 is writing.
Writer 34425 is writing.
Reader 34428 is reading.
Reader 34427 is reading.
Writer 34426 is writing.
^C
bramha@ubuntu:~/Desktop$ SS