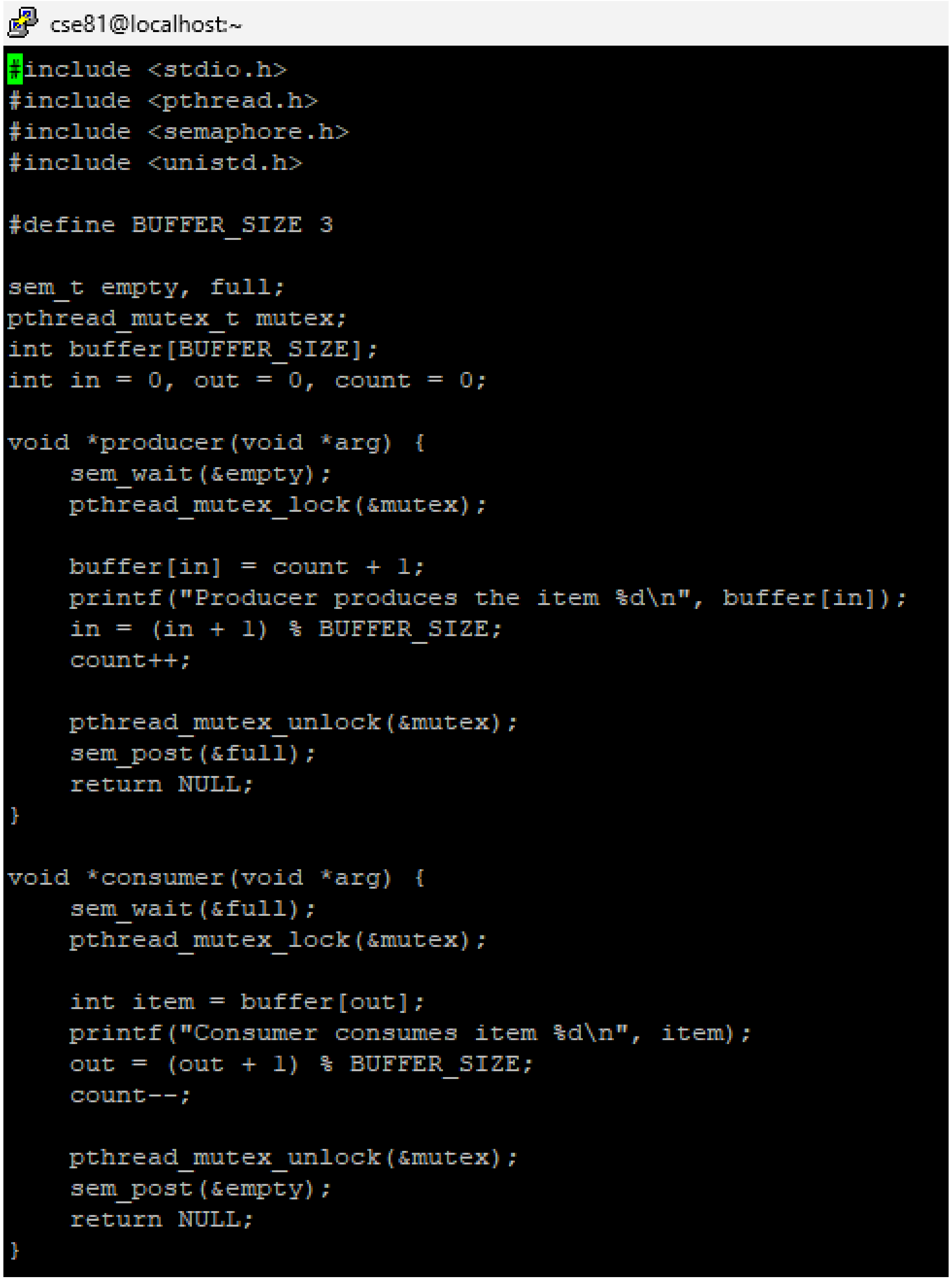
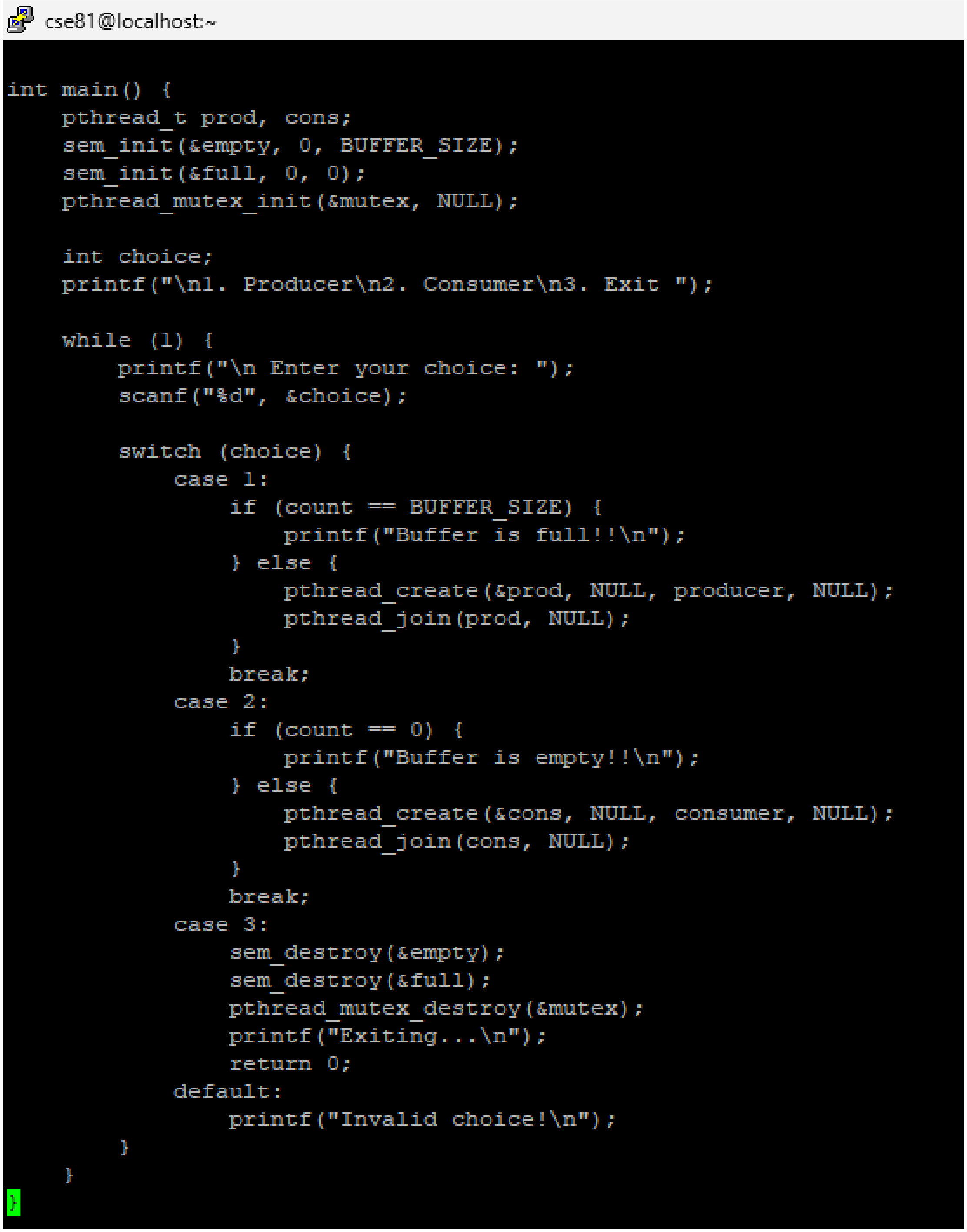
GIRIJA S P 230701092 CSE E

EXP NO:8 PRODUCER CONSUMER USING SEMAPHORES

PROGRAM

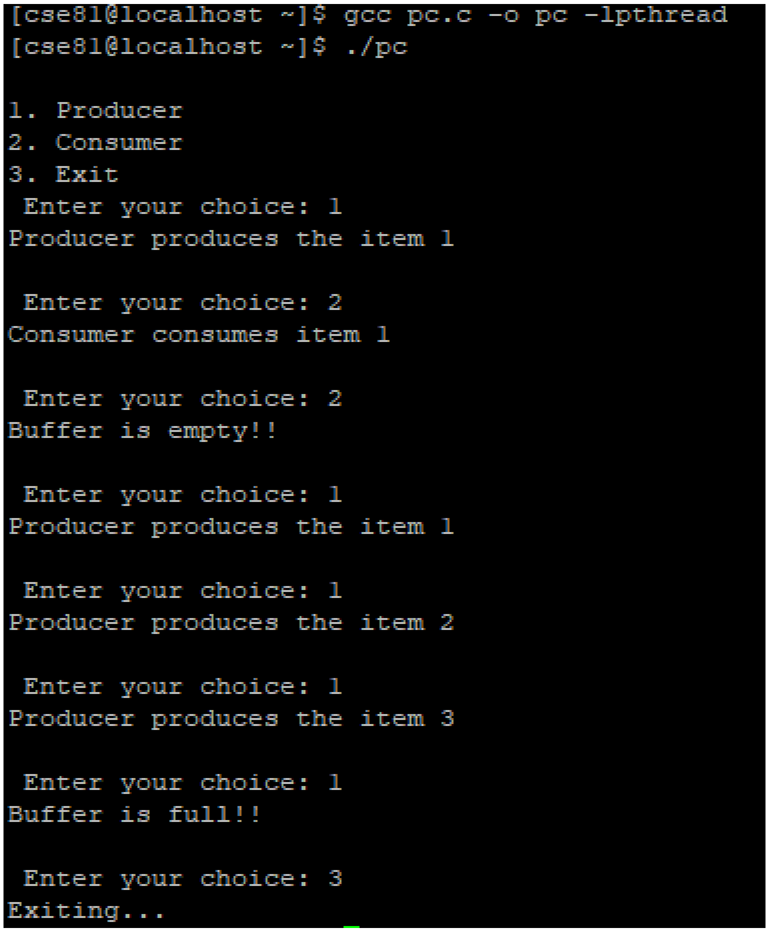


```
#include <stdio.h>
#include <pthread.h>
#include <semaphore.h>
#include <unistd.h>

#define BUFFER_SIZE 3

sem_t empty, full;
pthread_mutex_t mutex;
int buffer[BUFFER_SIZE];
int in = 0, out = 0, count = 0;

void *producer(void *arg) {
    sem_wait(&empty);
    pthread_mutex_lock(&mutex);

    buffer[in] = count + 1;
    printf("Producer produces the item %d\n", buffer[in]);
    in = (in + 1) % BUFFER_SIZE;
    count++;

    pthread_mutex_unlock(&mutex);
    sem_post(&full);
    return NULL;
}

void *consumer(void *arg) {
    sem_wait(&full);
    pthread_mutex_lock(&mutex);

    int item = buffer[out];
    printf("Consumer consumes item %d\n", item);
    out = (out + 1) % BUFFER_SIZE;
    count--;

    pthread_mutex_unlock(&mutex);
    sem_post(&empty);
    return NULL;
}
```



```
int main() {
    pthread_t prod, cons;
    sem_init(&empty, 0, BUFFER_SIZE);
    sem_init(&full, 0, 0);
    pthread_mutex_init(&mutex, NULL);

    int choice;
    printf("\n1. Producer\n2. Consumer\n3. Exit ");

    while (1) {
        printf("\n Enter your choice: ");
        scanf("%d", &choice);

        switch (choice) {
            case 1:
                if (count == BUFFER_SIZE) {
                    printf("Buffer is full!!\n");
                } else {
                    pthread_create(&prod, NULL, producer, NULL);
                    pthread_join(prod, NULL);
                }
                break;
            case 2:
                if (count == 0) {
                    printf("Buffer is empty!!\n");
                } else {
                    pthread_create(&cons, NULL, consumer, NULL);
                    pthread_join(cons, NULL);
                }
                break;
            case 3:
                sem_destroy(&empty);
                sem_destroy(&full);
                pthread_mutex_destroy(&mutex);
                printf("Exiting...\n");
                return 0;
            default:
                printf("Invalid choice!\n");
        }
    }
}
```

OUTPUT



```
[cse81@localhost ~]$ gcc pc.c -o pc -lpthread
[cse81@localhost ~]$ ./pc

1. Producer
2. Consumer
3. Exit
 Enter your choice: 1
Producer produces the item 1

 Enter your choice: 2
Consumer consumes item 1

 Enter your choice: 2
Buffer is empty!!

 Enter your choice: 1
Producer produces the item 1

 Enter your choice: 1
Producer produces the item 2

 Enter your choice: 1
Producer produces the item 3

 Enter your choice: 1
Buffer is full!!

 Enter your choice: 3
Exiting...
```