EXP 18: Construct a C program to simulate producer-consumer problem using semaphores.

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

#define SIZE 5

int buffer[SIZE];
int in = 0, out = 0;

sem_t empty, full;
pthread_mutex_t mutex;

void* producer(void* arg) {
  for (int i = 0; i < 10; i++) {
    int item = rand() % 100;

    sem_wait(&empty);
    pthread_mutex_lock(&mutex);

    buffer[in] = item;
    printf("Producer produced: %d\n", item);
    in = (in + 1) % SIZE;
```

```
        pthread_mutex_unlock(&mutex);
        sem_post(&full);

        sleep(1);
    }
    return NULL;
}

void* consumer(void* arg) {
    for (int i = 0; i < 10; i++) {
        sem_wait(&full);
        pthread_mutex_lock(&mutex);

        int item = buffer[out];
        printf("Consumer consumed: %d\n", item);
        out = (out + 1) % SIZE;

        pthread_mutex_unlock(&mutex);
        sem_post(&empty);

        sleep(2);
    }
    return NULL;
}

int main() {
    pthread_t prod, cons;
```

```
    sem_init(&empty, 0, SIZE);
    sem_init(&full, 0, 0);
    pthread_mutex_init(&mutex, NULL);

    pthread_create(&prod, NULL, producer, NULL);
    pthread_create(&cons, NULL, consumer, NULL);

    pthread_join(prod, NULL);
    pthread_join(cons, NULL);

    sem_destroy(&empty);
    sem_destroy(&full);
    pthread_mutex_destroy(&mutex);

    return 0;
}
```

Sample Output

```
Producer produced: 83
Consumer consumed: 83
Producer produced: 86
Consumer consumed: 86
Producer produced: 77
Producer produced: 15
Consumer consumed: 77
Producer produced: 93
Producer produced: 35
Consumer consumed: 15
Producer produced: 86
Producer produced: 92
Producer produced: 49
Consumer consumed: 93
Producer produced: 21
```

```
Consumer consumed: 35
Consumer consumed: 86
Consumer consumed: 92
Consumer consumed: 49
Consumer consumed: 21
```