EX –8 PRODUCER CONSUMER USING SEMAPHORES

NAME : Alfred Sam D

CLASS : CSE – A

ROLL NO : 230701026

Program:

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

#define BUFFER_SIZE 5

int buffer[BUFFER_SIZE];
int count = 0, in = 0, out = 0;

pthread_mutex_t mutex;
sem_t empty, full;

void producer() {
    if (count == BUFFER_SIZE) {
        printf("Buffer is full!!\n");
        return;
    }
```

```
    int item = in + 1;
    sem_wait(&empty);
    pthread_mutex_lock(&mutex);

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

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

void consumer() {
    if (count == 0) {
        printf("Buffer is empty!!\n");
        return;
    }

    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);
```

```
}
int main() {
    int choice;

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

    while (1) {
        printf("\nSample Output:\n");
        printf("1. Producer\n");
        printf("2. Consumer\n");
        printf("3. Exit\n");
        printf("Enter your choice:");
        scanf("%d", &choice);

        switch (choice) {
            case 1:
                producer();
                break;
            case 2:
                consumer();
                break;
            case 3:
                pthread_mutex_destroy(&mutex);
                sem_destroy(&empty);
                sem_destroy(&full);
                printf("Exiting...\n");
                exit(0);
```

```
        default:
            printf("Invalid choice! Please enter 1, 2, or 3.\n");
    }
  }

  return 0;
}
```

OUTPUT :