

### Assignment No. 4a

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

#define buffer 5
sem_t empty, full;
pthread_mutex_t mutex;
int in=0, out=0;

int arr[buffer];

// Helper function to print the buffer state
void print_buffer() {
    printf("Buffer: [");
    for (int i = 0; i < buffer; i++) {
        if (arr[i] == 0)
            printf("0");
        else
            printf("%d", arr[i]);
        if (i != buffer - 1) printf(", ");
    }
    printf("]\n");
}

void *producer(void *arg) {
    int item = 0;
    do{
        item++;
        sem_wait(&empty);
        pthread_mutex_lock(&mutex);

        arr[in] = item;
        printf("Producer produced %d at index %d\n", item, in);
        print_buffer();

        in = (in+1) % buffer;

        pthread_mutex_unlock(&mutex);
        sem_post(&full);
        sleep(1);
    }while(1);
    return NULL;
}

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

        item = arr[out];
        arr[out] = 0; // Mark slot as empty after consuming
```

```

        printf("Consumer consumed %d from index %d\n", item,
out);
        print_buffer();

        out = (out+1) % buffer;

        pthread_mutex_unlock(&mutex);
        sem_post(&empty);
        sleep(2);
    }while(1);
    return NULL;
}

int main(){
    pthread_t prod, cons;
    sem_init(&empty, 0, buffer);
    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);

    return 0;
}

```

```

manasvi@manasvi:/mnt/c/Users/bhute/Desktop/oslab/Ass 4$ gcc 4.c -o four
manasvi@manasvi:/mnt/c/Users/bhute/Desktop/oslab/Ass 4$ ./four
Producer produced 1 at index 0
Buffer: [1, 0, 0, 0, 0]
Consumer consumed 1 from index 0
Buffer: [0, 0, 0, 0, 0]
Producer produced 2 at index 1
Buffer: [0, 2, 0, 0, 0]
Consumer consumed 2 from index 1
Buffer: [0, 0, 0, 0, 0]
Producer produced 3 at index 2
Buffer: [0, 0, 3, 0, 0]
Producer produced 4 at index 3
Buffer: [0, 0, 3, 4, 0]
Consumer consumed 3 from index 2
Buffer: [0, 0, 0, 4, 0]
Producer produced 5 at index 4
Buffer: [0, 0, 0, 4, 5]
Producer produced 6 at index 0
Buffer: [6, 0, 0, 4, 5]
Consumer consumed 4 from index 3
Buffer: [6, 0, 0, 0, 5]
Producer produced 7 at index 1
Buffer: [6, 7, 0, 0, 5]
Producer produced 8 at index 2
Buffer: [6, 7, 8, 0, 5]
Consumer consumed 5 from index 4
Buffer: [6, 7, 8, 0, 0]
Producer produced 9 at index 3
Buffer: [6, 7, 8, 9, 0]

```