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

#define maxitems 5
#define buffersize 5

sem_t empty;
sem_t full;
int in=0;
int out=0;

int buffer[buffersize];
pthread_mutex_t  mutex;

void *producer(void *pno)
{
    int item;
    for(int i=0;i<maxitems;i++)
    {
        item=rand();
        sem_wait(&empty);
        pthread_mutex_lock(&mutex);
        buffer[in]=item;
        printf("Producer %d:insert item %d at %d \n",((int)pno),buffer[in],in);
        in=(in+1)%buffersize;
        pthread_mutex_unlock(&mutex);
        sem_post(&full);
    }
}

void *consumer(void *cno)
{
    for(int i=0;i<maxitems;i++)
    {
        sem_wait(&full);
        pthread_mutex_lock(&mutex);
        int item=buffer[out];
        printf("Consumer %d:Remove  item %d at %d \n",((int)cno),item,out);
        out=(out+1)%buffersize;
        pthread_mutex_unlock(&mutex);
        sem_post(&empty);
    }
}
int main()
{
    pthread_t pro[5],con[5];
    pthread_mutex_init(&mutex,NULL);
    sem_init(&empty,0,buffersize);
    sem_init(&full,0,0);

    int a[5]={1,2,3,4,5};

    for(int i=0;i<5;i++)
    {
        pthread_create(&pro[i],NULL,(void*)producer,(void*)&a[i]);
    }
    for(int i=0;i<5;i++)
    {
        pthread_create(&con[i],NULL,(void*)consumer,(void*)&a[i]);
    }

    for(int i=0;i<5;i++)
    {
        pthread_join(pro[i],NULL);
    }
    for(int i=0;i<5;i++)
    {
        pthread_join(con[i],NULL);
    }
    pthread_mutex_destroy(&mutex);
    sem_destroy(&empty);
    sem_destroy(&full);

    return 0;
}
```