```
#include<stdio.h>
#include<pthread.h>
#include<semaphore.h>
#include<stdlib.h>
sem_t s,empty,full;
int queue[5],avail;
void *producer(void);
void *consumer(void);

int main(void)
{
   pthread_t prod_h,cons_h;
   avail=0;
   sem_init(&s,0,1);
   sem_init(&empty,0,2);
   sem_init(&full,0,0);
   pthread_create(&prod_h,0,producer,NULL);
   pthread_create(&cons_h,0,consumer,NULL);
   pthread_join(prod_h,0);
   pthread_join(cons_h,0);
   exit(0);
}

void *producer(void)
{
   int prod=0;
   int item;
   while(prod<5)
   {
      item=rand()%1000;
      sem_wait(&empty);
      sem_wait(&s);
      queue[avail]=item;
      avail++;
      prod++;
      printf("The item produced in buffer %d \n",item);
      sleep(3);
      sem_post(&s);
      sem_post(&full);
      if(prod==5)
      {
         printf("Buffer is full \n");
      }
   }
   pthread_exit(0);
}

void *consumer(void)
{
   int cons=0,my_item;
   while(cons<5)
   {
      sem_wait(&full);
      sem_wait(&s);
      cons++;
      avail--;
      my_item=queue[avail];
```

```
        sem_post(&s);
        sem_post(&empty);
        printf("Consumed by %d: ",my_item);
        sleep(1);
        if(cons==0)
        {
           printf("Buffer is empty \n");
        }
    }
    pthread_exit(0);
}
```