#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);

}