Ex. No.: 8
Date:16.04.2025

PRODUCER CONSUMER USING SEMAPHORES

Aim:

To write a program to implement solution to producer consumer problem using semaphores.

Program:

```c
#include <stdio.h>
#include <stdlib.h>
#include <pthread.h>
#include <semaphore.h>
#define SIZE 3
int buffer[SIZE], in=0, out=0, count=0, item=0;
sem_t empty, full, mutex;

void *producer(){
    if(count==SIZE){
        printf("Buffer is full!!\n");
        return NULL;
    }
    sem_wait(&empty);
    sem_wait(&mutex);
    item++;
    buffer[in]=item;
    in=(in+1)%SIZE;
    count++;
    printf("Producer produces the item %d\n", item);
    sem_post(&mutex);
    sem_post(&full);
    return NULL;
}

void *consumer(){
    if(count==0){
        printf("Buffer is empty!!\n");
        return NULL;
    }
    sem_wait(&full);
    sem_wait(&mutex);
    int data=buffer[out];
    out=(out+1)%SIZE;
    count--;
    printf("Consumer consumes item %d\n", data);
    sem_post(&mutex);
    sem_post(&empty);
    return NULL;
}

int main(){
    sem_init(&empty,0,SIZE);
    sem_init(&full,0,0);
    sem_init(&mutex,0,1);
    int choice;
    while(1){
```

```c
printf("1.Producer\n2.Consumer\n3.Exit\nEnter your choice:");
scanf("%d",&choice);
pthread_t t;
if(choice==1)
    pthread_create(&t,NULL,producer,NULL);
else if(choice==2)
    pthread_create(&t,NULL,consumer,NULL);
else
    exit(0);
pthread_join(t,NULL);
    }
    return 0;
}
```

Output:

```
1.Producer
2.Consumer
3.Exit
Enter your choice:1
Producer produces the item 1
Enter your choice:2
Consumer consumes item 1
Enter your choice:2
Buffer is empty!!
Enter your choice:1
Producer produces the item 2
Enter your choice:1
Producer produces the item 3
Enter your choice:1
Producer produces the item 4
Enter your choice:1
Buffer is full!!
Enter your choice:3
```

Result:

The program to solve the producer-consumer problem using semaphores was executed successfully.