Ex. No.: 8
Date: 2 9/3/25

PRODUCER CONSUMER USING SEMAPHORES

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

Algorithm:

1. Initialize semaphore empty, full and mutex.
2. Create two threads- producer thread and consumer thread.
3. Wait for target thread termination.
4. Call sem_wait on empty semaphore followed by mutex semaphore before entry into critical section.
5. Produce/Consume the item in critical section.
6. Call sem_post on mutex semaphore followed by full semaphore
7. before exiting critical section.
8. Allow the other thread to enter its critical section.
9. Terminate after looping ten times in producer and consumer Threads each.

Program Code:

```
#include < stdio.h>
# include <stdlib.h>
# include < semaphore.h>
# include < unistd.h>
# define BUFFER_SIZE 3
 int buffer [BUFFER_SIZE];
int cont =0;
 Semt empty;
 Semt full;
 Semt mutex;
void producer() {
        Sem_wait (& empty);
        Sem_wait (& mutex);
if ( count < BUFFERSIZE) {
```

```
        if (count < BUFFER SIZE) {
            int item;
            printf("Enter item to produce");
            scanf("%d", &item);
        buffer[count] = item;
        printf("Producer produced item %d\n", item);
        count++;
    } else {
        printf("Buffer is full\n");
    } sem_post(&mutex);
      sem_post(&full);
}
void consumer() {
        sem_wait(&full); sem_wait(&mutex);
        if (count > 0) {
            int item = buffer[count-1];
            printf("Consumer consumes item %d\n", item);
            count--;
    } else {
        printf("Buffer is empty");
    } sem_post(&mutex); sem_post(&empty);
}
int main() {
    int choice;
    sem_init(&empty, 0, BUFFER SIZE);
    sem_init(&full, 0, 0);
    sem_init(&mutex, 0, 1);
    while(1) {
```

```
printf("\n1. Producer \n2. Consumer\n3. Exit\n Enter your choice:");
scanf("%d", &choice);
switch (choice) {
    case 1:
        producer(); break;
    case 2:
        consumer(); break;
    case 3:
        printf("Exiting...");
        sem_destroy(&empty);
        sem_destroy(&full);
        sem_destroy(&mutex); return 0;
    default:
        printf("Invalid choice\n");
    }
}
```

Sample 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 1
Enter your choice:1
Producer produces the item 2
Enter your choice:1
Producer produces the item 3
Enter your choice:1
Buffer is full!!
Enter your choice:3

O/P

1. Producer
2. Consumer
3. Exit

Enter your choice : 1
Enter item to produce : 1
~~Enter item~~ .
producer produces item : 1
Enter your choice : 1
Enter item to produce : 2
producer produce item 2
Enter your choice : 2
Consumer consumes item 2
Enter your choice : 3
Exiting....

Result:

Hence the program to implement solution to producer consumer problem using semphores has been successfully executed.