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 criticalsection.
5. Produce/Consumer the item in critical section.
6. Call sem_post on mutex semaphore followed by full semaphore before exiting critical section.
7. Allow the otherthread to enter its criticalsection.
8. Terminate after looping ten times in producer and consumer threads each.

Program Code:

```
#include<stdio.h>
#include<stdlib.h>
int mutex=1,full=0,empty=3,x=0;
int main()
{
        int n;
        void producer();
        void consumer();
        int wait(int);
        int signal(int);
        printf("\n1.Producer\n2.Consumer\n3.Exit");
        while(1)
        {
                printf("\nEnter your choice:");
                scanf("%d",&n);
                switch(n)
                {
                        case 1:
                        if((mutex==1)&&(empty!=0))
                                producer();
```

```
                        else
                                printf("Buffer is full!!");
                        break;
                        case 2:
                        if((mutex==1)&&(full!=0))
                                consumer();
                        else
                                printf("Buffer is empty!!");
                        break;
                        case 3:
                                exit(0);
                                break;
                }
        }
        return 0;
}
int wait(int s)
{
        return (--s);
}
int signal(int s)
{
        return(++s);
}
void producer()
{
        mutex=wait(mutex);
        full=signal(full);
        empty=wait(empty);
        x++;
        printf("\nProducer produces the item %d",x);
        mutex=signal(mutex);
}
void consumer()
{
        mutex=wait(mutex);
        full=wait(full);
        empty=signal(empty);
        printf("\nConsumer consumes item %d",x);
        x--;
        mutex=signal(mutex);
}
```