PAGE NO 14/7/23
DATE :

Write a C program to simulate producer, consumer problem using semaphore

```
#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("\n producer produces the item %d", x);
     mutex = Signal (mutex);
}
Void consumer()
{
  mutex = wait (mutex);
   full = wait (full);
  empty = signal (empty);
    printf ("\n consumer Consumes item %d", x)
     x--;
    mutex = Signal (mutex);
}
```

Output

1. producer 2. consumer 3. Exit.
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: 2
Consumer consumes item 2
Enter your choice: 2
Consumes Consumes item 1
Enter your choice: 2
Buffer is empty !!

10/10

21/7/23

```
Producer produces item: 1
Consumer consumes item: 1
Producer produces item: 1
Consumer consumes item: 1
Producer produces item: 1
Consumer consumes item: 1
Producer produces item: 1
Consumer consumes item: 1
Producer produces item: 1
Consumer consumes item: 1
Producer produces item: 1
Consumer consumes item: 1
Producer produces item: 1
Consumer consumes item: 1
Producer produces item: 1
```