

Ex. No.: 8**Date:****19.03.25****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>

int mutex = 1;
int full = 0;
int empty;
int x = 0;

void producer() {
    mutex--;
    full++;
    empty--;
    x++;
    printf("\nProducer produces item %d", x);
    mutex++;
}

void consumer() {
    --mutex;
```

```
--full;
++empty;
printf("\nConsumer consumes item %d", x);
x--;
++mutex;
}

int main() {
    int n, i;
    printf("\n1. Press 1 for producer\n2. Press 2 for consumer\n3. Press 3 for exit");
    for (i = 1; i > 0; i++) {
        printf("\nEnter your choice: ");
        scanf("%d", &n);
        empty = 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;
        }
    }
}
```

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

Output:

```
(student@kali)-[~]  
$ vi procon.c  
  
(student@kali)-[~]  
$ gcc procon.c -o procon  
  
(student@kali)-[~]  
$ ./procon  
  
1.Press  for producer  
2.Press 2 for consumer  
3.Press 3 for exit  
Enter your choice:1  
  
Producer produces item 1  
Enter your choice:1  
  
Producer produces item 2  
Enter your choice:1  
  
Producer produces item 3  
Enter your choice:1  
  
Producer produces item 4  
Enter your choice:2  
  
Consumer consumes item 4  
Enter your choice:3
```

Result:

Thus,the producer consumer program has been successfully executed.