

## SPOT EXERCISE

### LAB 10

P.PRIYADHARSHINI

2019103562

12-04-2021

```
[s2019103562@centos8-linux Mon Apr 12 12:13 PM lab10]$ ./a.out
1.PRODUCER
2.CONSUMER
3.EXIT
Enter choice:1
Produced an item 1
1.PRODUCER
2.CONSUMER
3.EXIT
Enter choice:1
Produced an item 2
1.PRODUCER
2.CONSUMER
3.EXIT
Enter choice:1
Produced an item 3
1.PRODUCER
2.CONSUMER
3.EXIT
Enter choice:1
Produced an item 4
1.PRODUCER
2.CONSUMER
3.EXIT
Enter choice:1
Produced an item 5
1.PRODUCER
2.CONSUMER
3.EXIT
Enter choice:1
Produced an item 6
1.PRODUCER
2.CONSUMER
3.EXIT
Enter choice:1
BUFFER IS FULL
1.PRODUCER
2.CONSUMER
3.EXIT
Enter choice: 
```

```
2.CONSUMER
3.EXIT
Enter choice:2
Consumed an item 6
1.PRODUCER
2.CONSUMER
3.EXIT
Enter choice:2
Consumed an item 5
1.PRODUCER
2.CONSUMER
3.EXIT
Enter choice:2
Consumed an item 4
1.PRODUCER
2.CONSUMER
3.EXIT
Enter choice:2
Consumed an item 3
1.PRODUCER
2.CONSUMER
3.EXIT
Enter choice:2
Consumed an item 2
1.PRODUCER
2.CONSUMER
3.EXIT
Enter choice:2
Consumed an item 1
1.PRODUCER
2.CONSUMER
3.EXIT
Enter choice:2
BUFFER IS EMPTY
1.PRODUCER
2.CONSUMER
3.EXIT
Enter choice:3
Enter valid choice
[s2019103562@centos8-linux Mon Apr 12 12:13 PM lab10]$
```

```
[s2019103562@centos8-linux Mon Apr 12 12:13 PM lab10]$ cat spot1.c
#include<stdio.h>
#include<stdlib.h>
#include<semaphore.h>
#define BUFFERSIZE 6
sem_t mutex,full,empty;
void producer(int a){
    sem_wait(&empty);
    sem_wait(&mutex);
    printf("Produced an item %d\n",a);
    sem_post(&mutex);
    sem_post(&full);
}
void consumer(int b){
    sem_wait(&full);
    sem_wait(&mutex);
    printf("Consumed an item %d\n",b);
    sem_post(&mutex);
    sem_post(&empty);
}
int main(){
    int opt;
    int mu,empt,ful;
    sem_init(&mutex,0,1);
    sem_init(&full,0,0);
    sem_init(&empty,0,BUFFERSIZE);
    int a=1;
    do{
        printf("1.PRODUCER\n");
        printf("2.CONSUMER\n");
        printf("3.EXIT\n");
        printf("Enter choice:");
        scanf("%d",&opt);
        sem_getvalue(&mutex,&mu);
        sem_getvalue(&empty,&empt);
        sem_getvalue(&full,&ful);
```

```
        switch(opt){
            case 1:
                if(mu==1 && empt!=0){
                    producer(a);
                    a++;
                }
                else
                    printf("BUFFER IS FULL\n");
                break;
            case 2:
                if(mu==1 && ful!=0){
                    a--;
                    consumer(a);
                }
                else
                    printf("BUFFER IS EMPTY\n");
                break;
            default:
                printf("Enter valid choice\n");
        }
    }while(opt!=3);
    return 0;
}
[s2019103562@centos8-linux Mon Apr 12 12:14 PM lab10]$
```