EXPERIMENT NO. 4

AIM: Write a program to implement the solution of Producer Consumer problem through Semaphore.

RESOURCES REQUIRED:

H/W Requirements: P-IV and above, Ram 128 MB, Printer, Internet Connection.

S/W Requirements: Python Compiler.

THEORY:

THE PRODUCER CONSUMER PROBLEM:

- There are one or more producers generating some type of data (records, characters) and placing these in a buffer.
- There is a single consumer that is taking items out of the buffer one at a time.
- The system is to be constrained to prevent the overlap of buffer operations. That is, only one agent (producer or consumer) may access the buffer at any one time.
- The problem is to make sure that the producer won't try to add data into the buffer if it's full, and that the consumer won't try to remove data from an empty buffer.

To begin, let us assume that the buffer is infinite and consists of a linear array of elements. In abstract terms, we can define the producer and consumer functions as follows:

```
producer:
while (true) {
     /* produce item v */;
     b[in] = v;
     in++;
}
```

```
consumer:
while (true) {
    while (in <= out)
        /* do nothing */;
    w = b[out];
    out++;
    /* consume item w */;
}
```

CONCLUSION: Hence, we have implemented a program on Producer Consumer problem through semaphore.

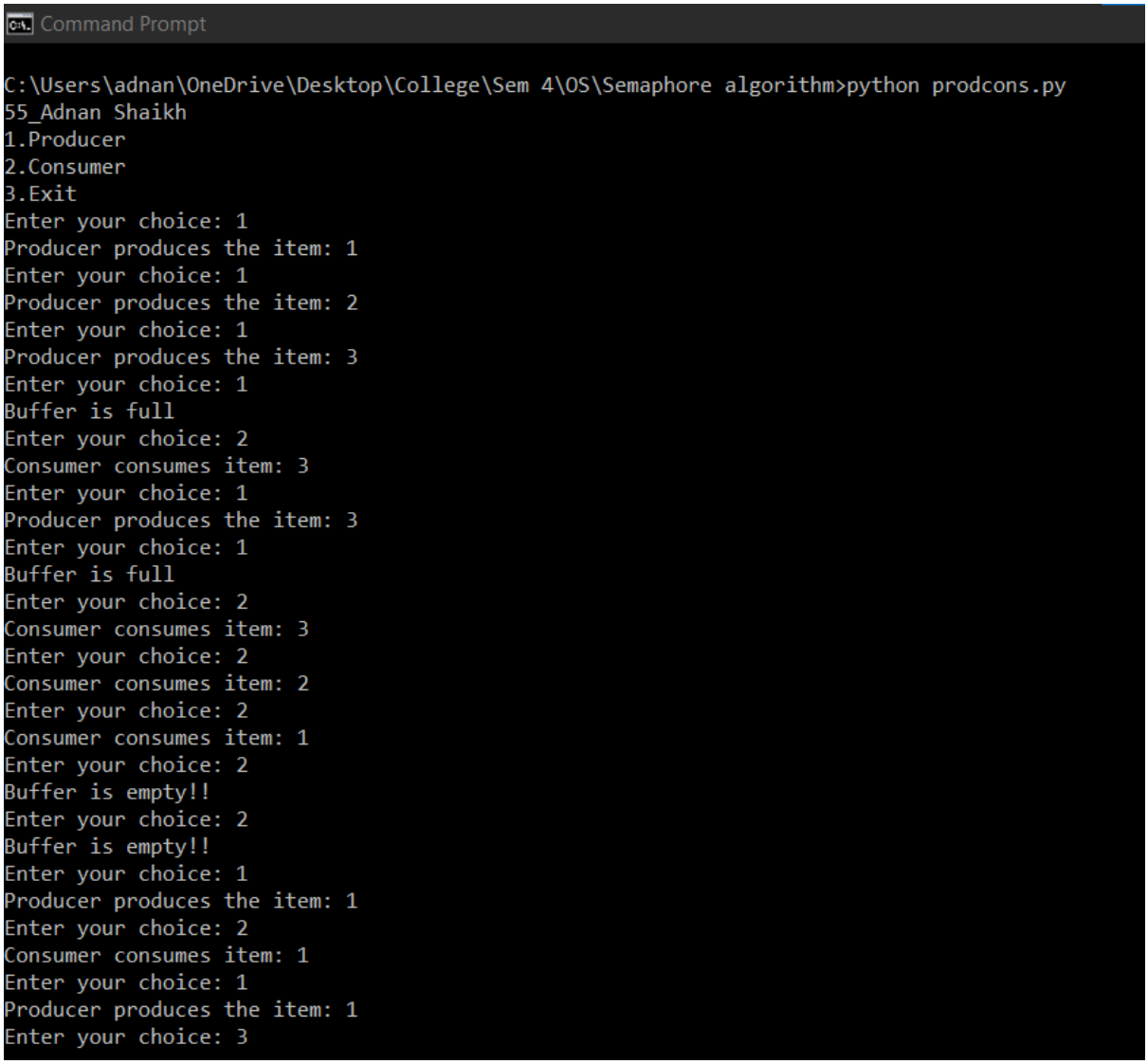
CODE:

```
mutex,full,empty,x = 1,0,3,0
def wait(s):
    global mutex,full,empty,x
    s -=1
    return s

def signal(s):
    global mutex,full,empty,x
    s += 1
    return s
```

```
def producer():
    global mutex,full,empty,x
    mutex = wait(mutex)
    full = signal(full)
    empty = wait(empty)
    x +=1
    print("Producer produces the item: "+str(x))
    mutex = signal(mutex)

def consumer():
    global mutex,full,empty,x
    mutex = wait(mutex)
    full = wait(full)
    empty = signal(empty)
    print("Consumer consumes item: "+str(x))
    x -=1
    mutex = signal(mutex)

if __name__ == "__main__":
    print("55_Adnan Shaikh")
    print("1.Producer \n2.Consumer \n3.Exit")

    while(1):
        n = int(input("Enter your choice: "))

        if n == 1:
```

```
    if mutex == 1 and empty != 0:
        producer()
    else:
        print("Buffer is full")
elif n == 2:
    if  mutex == 1 and full != 0:
        consumer()
    else:
        print("Buffer is empty!!")
elif n == 3:
    break
else:
    print("You pressed wrong key please try")
```

OUTPUT:


Command Prompt
C:\Users\adnan\OneDrive\Desktop\College\Sem 4\OS\Semaphore algorithm>python prodcons.py
55_Adnan Shaikh
1.Producer
2.Consumer
3.Exit
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: 2
Consumer consumes item: 3
Enter your choice: 1
Producer produces the item: 3
Enter your choice: 1
Buffer is full
Enter your choice: 2
Consumer consumes item: 3
Enter your choice: 2
Consumer consumes item: 2
Enter your choice: 2
Consumer consumes item: 1
Enter your choice: 2
Buffer is empty!!
Enter your choice: 2
Buffer is empty!!
Enter your choice: 1
Producer produces the item: 1
Enter your choice: 2
Consumer consumes item: 1
Enter your choice: 1
Producer produces the item: 1
Enter your choice: 3