Ex. No.: 8
Date: 02/04/25

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:

```
import threading
import time
import random

buffer = [ ]
BUFFER_SIZE = 5

empty = threading.semaphore(BUFFER_SIZE)
full = threading.semaphore(0)
mutex = threading.semaphore(1)

def producer():
        item = random.randint(1,10)
        if empty.acquire(blocking=False):
                mutex.acquire()
                buffer.append(item)
                print(f"Producer produces item
                {item}")
                mutex.release()
                full.release()
        else:
                print("Buffer is full. Producer can't produce
                right now.")
```

```
def consumer():
        if full.acquire(blocking=False):
                mutex.acquire()
                item = buffer.pop()
                Printf"consumer consumes item {item}")
                mutex.release()
                empty.release()
        else:
            print("Buffer is empty. consumer can't
             consume right now.")

while True:
        print("\nMenu: ")
        print("1.Producer")
        print("2.consumer")
        print("3. Exit")
        choice = input("Enter your choice: ")
        if choice == '1':
                producer()
        elif choice == '2':
                consumer()
        elif choice == '3':
                print("Exiting...")
        else:
                print("Invalid choice. Please enter 1,2 or 3.")
```

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:

Menu:
1. producer
2. consumer
3. Exit
Enter your choice: 1
producer produces item 8

Menu:
1. producer
2. consumer
3. Exit
Enter your choice: 2
consumer consumes item 8.

Menu:
1. producer
2. consumer
3. Exit
Enter your choice: 2
Buffer is empty. consumer can't consume right now.

Result:

Therefore the program to implement solution to producer consumer problem using semaphores was executed successfully.