

Section A**1. Producer-Consumer problem:**

Aim: To write a c program to simulate the producer consumer problem using semaphores

Description:

Producer consumer problem is a synchronization problem. There is a fixed size buffer where the producer produces items and that is consumed by a consumer process. One solution to the producer- consumer problem uses shared memory. To allow producer and consumer processes to run concurrently, there must be available a buffer of items that can be filled by the producer and emptied by the consumer. This buffer will reside in a region of memory that is shared by the producer and consumer processes. The producer and consumer must be synchronized, so that the consumer does not try to consume an item that has not yet been produced.

```
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: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:_
```

Producer Consumer Sample Run

Section B

You are requested to implement 2 of the scheduling algorithms explained in the previous lab namely:

1. FCFS
2. SJF

but with two modifications.

1. The arrays used should be dynamic.
2. The arrival time should be taken into consideration (Not equal to 0).

Hint:

Turnaround Time = Completion Time – Arrival Time

Waiting Time = Turnaround Time – Burst Time

```
Microsoft Visual Studio Debug Console
Enter the number of processes -- 5
Enter burst time for processes 0 -- 5
Enter arrival time for processes 0 -- 4
Enter burst time for processes 1 -- 4
Enter arrival time for processes 1 -- 6
Enter burst time for processes 2 -- 3
Enter arrival time for processes 2 -- 0
Enter burst time for processes 3 -- 2
Enter arrival time for processes 3 -- 6
Enter burst time for processes 4 -- 4
Enter arrival time for processes 4 -- 5

Average waiting time = 4.400000
Average turnaround time = 8.000000

Process    Arrival Time    Burst Time    Waiting Time    Turnaround Time
P3         0              3             0              3
P1         4              5             0              5
P5         5              4             4              8
P2         6              4             7             11
P4         6              2            11             13
```

FCFS Sample Run

```
Microsoft Visual Studio Debug Console
Enter the number of processes -- 4
Enter burst time for processes 0 -- 7
Enter arrival time for processes 0 -- 0
Enter burst time for processes 1 -- 4
Enter arrival time for processes 1 -- 2
Enter burst time for processes 2 -- 1
Enter arrival time for processes 2 -- 4
Enter burst time for processes 3 -- 4
Enter arrival time for processes 3 -- 5

Average waiting time = 4.000000
Average turnaround time = 8.000000

Process    Arrival Time    Burst Time    Waiting Time    Turnaround Time
P0         0              7             0              7
P2         4              1             3              4
P1         2              4             6             10
P3         5              4             7             11
```

SJF Sample Run