

**University of Engineering and Technology, Lahore**  
**Operating Systems Lab.**

**Semester: 6th**

**Session: 2019-23**

**Lab-08 (Individual)**

**Submission Date:** -03-22 (Before 12:00 pm)

**Note:** Understanding the assignment is part of the assignment.

**Objective:**

To learn synchronization through Semaphores between threads

**Question # 1**

There are exactly 3 threads generate string a, b and c in an arbitrary order. In an absence of any synchronization mechanism there will be no order in generation of a, b and c. Synchronize threads using semaphore in such a way that your printed string will be aaacbaaacbaacb.....

<pre>//thread 1 While(1) {     cout &lt;&lt; 'a'; }</pre>	<pre>//thread 2 While(1) {     Cout &lt;&lt; 'b'; }</pre>	<pre>//thread 3 While(1) {     Cout &lt;&lt; 'c'; }</pre>
---	---	---

**Question # 2**

Two Threads i.e. Producer and Consumer are sharing a buffer of size 100. Producer generates random number from 1 to 6, stores that number in the buffer and print the number. Consumer has to read values from the buffer added by the producer, perform the summation operation and display the result. Producer will stop after adding 100 values. Both producer and consumer threads run simultaneously; hence, synchronize.

You're not allowed to use cout/printf statements

The output should look Like

Number 2 //printed by producer

Number 5

Number 4

Number 2

Number 6

Number 2

Number 5

Number 1

Number 4

Number 2

Number 3

Number 4

Number 3

Number 5

Sum 2 //printed by consumer

Sum 7

Number 2

Sum 11

Sum 13