Lab Assignment 5

**Title: Process synchronization**

**Problem statement:**

To implement Producer-Consumer problem

**Objectives:**

* To understand the concept of Process Synchronization
* To understand the classical Producer-Consumer problem
* To devise a solution using semaphores

**Theory:**

Producer-Consumer problem is one of the classic problems of synchronization. Producer produces an item and adds to a buffer of limited size (bounded buffer). Consumer takes out an item from buffer and consumes it. Buffer is a shared resource used as a queue and must be accessed in a mutual exclusion manner by both processes. Producers to be prevented from adding into a full buffer. Consumers to be stopped from taking out an item from an empty buffer

Solution to producer-consumer problem using semaphore:

The bounded buffer producer problem assumes that there is a fixed buffer size.

In this case, the consumer must wait if the buffer is empty and the producer must wait if the buffer is full.

Initialization

char item; //could be any data type

char buffer[n];

semaphore full = 0; //counting semaphore for full slots

semaphore empty = n; //counting semaphore for empty slots

semaphore mutex = 1; //binary semaphore for mutual exclusion of buffer

char nextp, nextc;

***Producer Process***

*do*

*{*

*produce an item in nextp*

*wait (empty);*

*wait (mutex);*

*add nextp to buffer*

*signal (mutex);*

*signal (full);*

*} while (true)*

***Consumer Process***

*do*

*{*

*wait (full);*

*wait (mutex);*

*remove an item from buffer to nextc*

*signal (mutex);*

*signal (empty);*

*consume the item in nextc;*

*} while (true)*

**Conclusion**: Thus, we have studied and implemented the concept of process syncronization

**FAQs**

1. Explain the working of semaphores.
2. Discuss reader-writer problem and devise a solution using semaphores.
3. List and discuss the different process synchronization mechanisms.