

CSD 204 Lab

Lab 5: Process Synchronization

[Dr. Sweta Kumari, Assistant Professor, SNioE]

Deadline – 9th Mar 2025, 11:59PM

Problem Statement: Producer-Consumer Problem

Solve the classic **Producer-Consumer problem** using synchronization techniques in C++.

In this problem, two types of processes, **producers** and **consumers**, share a common buffer of fixed size. The **producer** generates data and inserts it into the buffer, while the **consumer** consumes data from the buffer. The challenge is to ensure that the producer does not produce data when the buffer is full and that the consumer does not consume data when the buffer is empty.

Objective: Implement a program using semaphores to solve the producer-consumer problem, ensuring that the producer and consumer processes operate correctly under synchronization constraints.

Test Case 1: Producer-Consumer with Buffer Size 5

Input:

- Buffer size = 5
- Producer thread produces data (randomly generated)
- Consumer thread consumes data (consumes data from the buffer)

Expected Output:

- The producer should only produce when there is space in the buffer.
- The consumer should only consume when there is data in the buffer.
- The output will alternate between producers adding items and consumers consuming items.

Producer 1 produced: 23

Producer 2 produced: 45

Consumer 1 consumed: 23

Consumer 2 consumed: 45

...

Test Case 2: Full Buffer (Producer Blocking)

Input:

- Buffer size = 3
- Producer thread generates data and attempts to insert it into the buffer.
- Consumer thread consumes data from the buffer.

Steps:

1. Producer produces 3 items (buffer reaches capacity).
2. Consumer consumes 2 items.
3. Producer attempts to add another item (should block until space is available).

Expected Output:

- After the buffer reaches full capacity, the producer should block until the consumer consumes at least one item and frees up space in the buffer.

Test case 3: Continuous Operation (Multiple Producers & Consumers)

Input:

- Buffer size = 5
- Multiple producer and consumer threads running concurrently.

Expected Output:

- Producers should produce when there is space, and consumers should consume when data is available.
- The output will alternate between producers and consumers, and the system should avoid any deadlocks or race conditions.

Producer 1 produced: 5

Producer 2 produced: 10

Consumer 1 consumed: 5

Consumer 2 consumed: 10

Producer 1 produced: 15

Producer 2 produced: 20

...

Complementary Question 2: Implement a program using semaphores to solve the printer-spooler problem, ensuring that the printer and spooler processes operate correctly under synchronization constraints.

Submission Format:- You have to upload: (1) The source code in the following format: Assgn5Src-<Name>.c (2) Report: Assgn5Report-<Name>.pdf. Name the zipped document as: Assgn5-<Name>.zip.

Note: Please follow this naming convention mentioned above. make different files for codes for each question.

Grading Policy:- The policy for grading this assignment will be - (1) Design as described in the report and analysis of the results: 10% (2) Code Execution and indentation of the tasks 90%

Please note:

- All assignments for this course have a late submission policy of a penalty of 10% each day after the deadline of six days. After that, it will not be evaluated.
- **All submissions are subject to plagiarism checks. Any case of plagiarism will be dealt with severely.**