Module 3 : Process Concurrency

★ Concurrency encompasses issues including –
- communication among processes.
- sharing and competing for resources
- synchronization
- allocation of processor time to process.

★ Principles of Concurrency –

1] The activities of other processes
2] The way OS handles interrupts
3] Scheduling policies of the OS.

★ Aspects of Concurrency Control –

1] Mutual Exclusion – to share code, resources or data in such a way that only one process has access to the shared object at a time.

2] Synchronization – ability of multiple processes to coordinate their activities by the exchange of information

3] Deadlock – situation where two or more processes are unable to proceed because each of the other are waiting for others to do something.

4] Starvation - situation in which a runnable process is overlooked by the scheduler, even if its able to proceed, it is never chosen.

5] Livelock - situation where two or more processes continuously change their states in response to changes in other processes without doing any useful work.

6] Race Condition - situation in which multiple threads or process compete for a shared/common resource and final result depends on the relative timing of their execution.

7] Critical Section - piece of code in which the shared resource is accessed by the process.

8] Data Coherence - consistency of data or variables stored in shared memory.

* Classic Problems in Concurrency

1] Reader/Writer's Problem
2] Producer Consumer Problem
3] Dining Philosopher Problem.

Page No.
Date

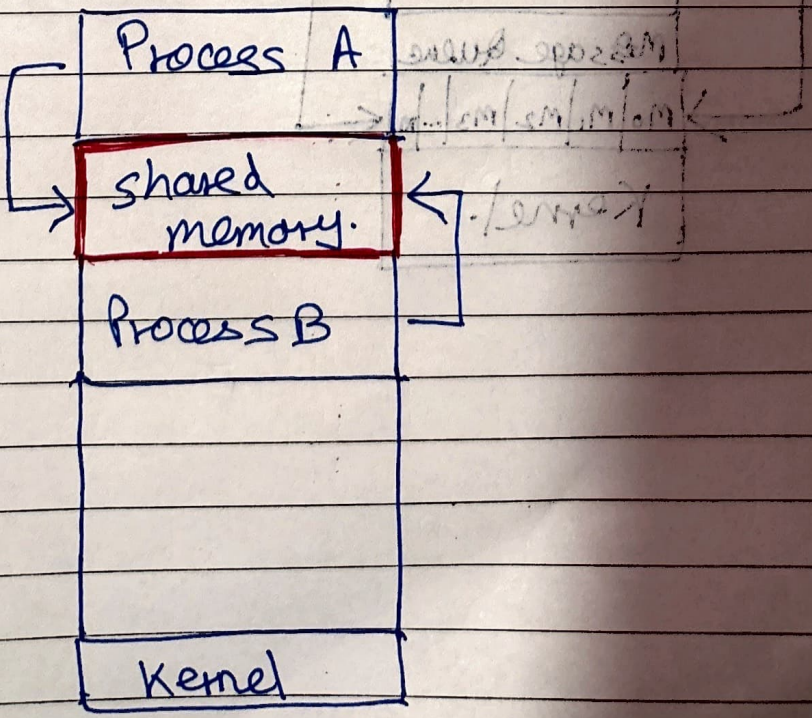
★ Interprocess Communication (IPC)

Coorepating process needs ↑ interprocess communication mechanism to exchange data & information.

Two Models of IPC :-

1] Shared Memory Model

Region of memory that is shared by cooperating processes is established.
Processes can exchange info by reading and writing data to the shared region
Allows max speed & convenience of communication
Faster than message passing
Once established, all accesses are routine memory accesses, no assistance from kernel is required.



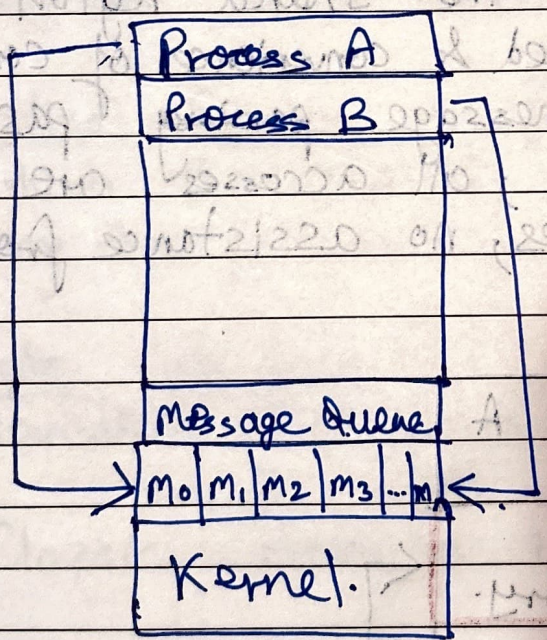
2] Message Passing Model

Communication takes place by means of messages exchanged between cooperating processes.

Useful for exchanging smaller amount of data; because no conflicts need to be avoided.

Also easier to implement than shared memory in intercomputer communication.

Slow process since system calls are used and kernel intervention.



✱ Process Synchronization.

Processes can execute concurrently but may be interrupted at any time, partially completing execution

Concurrent access to shared data may result in data inconsistency

Concurrency is achieved by using multiple processors, cores, threads depending on level of parallelism.

However, Whereas,

Synchronization is the coordination of concurrent tasks to ensure they do not interfere with each other or access shared resources in an inconsistent or unsafe way.