Q3. What are parallel programming model?

→ A programming model is a collection of program abstractions providing programmer a simplified & transparent view of computer hardware / software system

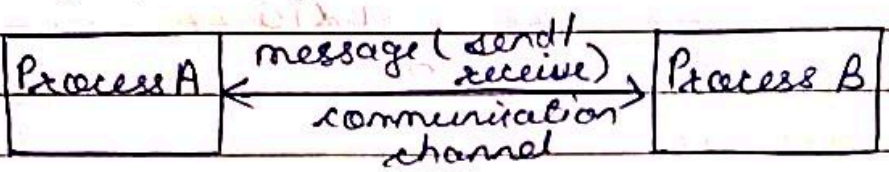
There are five parallel programming models -

1] shared - variable model
2] message passing model
3] data - parallel model
4] object oriented model
5] functional & logic model

1] shared - variable model

- A] shared - variable communication → Multiprocessor programming is based on use of shared variables in common memory for IPC.
- B] critical section → A critical section is code segment accessing shared variables which must be executed by only one process at a time & which once started must be completed without interruption.
- C] Partitioning & replication → Program partitioning is a technique for decomposing a large program & data set into many small pieces for parallel execution by multiple processors. Program replication refers to duplication of same program code for parallel execution on multiple processors over different data sets.

2] Message passing model-



- Two processes A & B residing at different processor nodes may communicate with each other by passing messages through a direct or indirect network.
- Synchronous msg passing → must synchronize the sender process & the receiver process in time & space.
- Asynchronous msg passing → does not require msg sending & receiving be synchronized in time & space.

3] Data parallel model

- The data parallel code is easier to write & to debug because parallelism is explicitly handled by hardware synchronization & flow control.
- Data parallel languages are modified directly from std serial programming languages.
- It can be implemented either on SIMP computer or on SPMD multicomputer, depending on the grain size & operation mode adopted.
- Array language extension → Array extensions in data parallel languages are ~~supported~~ represented by high level data types. The array syntax enables the removal of some nested loops in the code & should reflect the architecture of array processor. Ex - CFD for illiac IV, DAP FORTRAN.

- compiler support → To support data-parallel programming, the array lang. expressions & their optimizing compiler must be embedded in familiar std such as fortran 77, fortran 90 & C.

4] Object-oriented model -
- In this model, objects are dynamically created & manipulated. Processing is performed by sending & receiving msg over internet.
- concurrent OOP → The development of concurrent OOP provides an alternative model for concurrent computing on multiprocessors or on multicomputers. Various object models differ in the internal behavior of objects & in how they interact with each other.
- pallelism parallelism in COOP -
  ① pipeline concurrency → involves overlapped enumeration of successive solutions
  ② divide & conquer concurrency → involves concurrent elaboration of different subprograms
  ③ cooperative problem solving.

5] Functional & logical model -
- Functional model → emphasizes the functionality of a program & should not produce side effects after execution. There is no concept of storage, assignment & branching in functional programs.
- logical model → Logic programming is suitable for knowledge processing dealing with large databases. This model adopts an implicity search strategy & supports parallelism in the logic inference process.