IT801-N: Distributed and Parallel Computing

Unit 1. Introduction [Approx. Marks 5/70]

1.1. Why do we need more computational power? Explain with any one real-time example.
1.2. Differentiate UMA Vs. NUMA.
1.3. Differentiate Multiprocessor Vs. Multicomputer
1.4. List various topologies used for message passing system. Explain any one of them with neat diagram.
1.5. Explain message passing multicomputer in detail [5, April 2022]
1.6. Explain Flynn's classification / taxonomy. OR. Differentiate SIMD Vs. MIMD.
1.7. Explain Cluster computing.

Unit 2. Message Passing Computing [Approx. Marks 7/70]

2.1. Explain the following programming model with neat diagram.
(a) Single-program Multiple-data Model (SPMD) (b) Multiple-program Multiple-data Model (MPMD)
2.2. What is "spawning a process"?
2.3. Explain synchronous message passing.
2.4. Explain Static and Dynamic Process Creation in Message-Passing Programming [5, April 2022]
2.5. What is the need of "buffer" in message passing?
2.6. Explain following with reference to message passing among multiple processes.
(a) Broadcast (b) Scatter (c) Gather (d) Reduce
2.7. How space-time diagram can be used as a visualization tool?
2.8. How execution time (or elapsed time) is measured?
2.9. What is a profile? How does it useful?

Unit 3. Partitioning and Divide-and-Conquer Strategies [Approx. Marks 7/70]

3.1. List the partitioning strategies and explain them in brief.
3.2. Explain the partitioning with the example of adding 'n' numbers.
3.3. Explain the communication and computation phases of partitioning.
3.4. Explain divide-and-conquer in brief.
3.5. Write the algorithm for divide-and-conquer.
3.6. Explain M-ary Divide and Conquer.

Unit 4. Pipelined Computations [Approx. Marks 10/70]

4.1. Explain pipeline processing with time-space diagram [5, April 2022]
4.2. Explain (a) Pipeline for an unrolled loop OR (b) Pipeline for a Frequency Filter with neat diagram.
4.3. Explain the usage of Pipeline with space-time diagram.
4.4. Explain with neat diagram, how 10 elements of an array are multiplied using pipelining where individual elements enter the pipeline as sequential series of numbers.
4.5. Explain with diagram, how multiprocessor system can be used to add 'n' numbers.
4.6. Explain sorting with pipelining. (Take any 5 sample numbers of your choice to demonstrate the process)

Unit 5. Synchronous Computations [Approx. Marks 12/70]

5.1. Explain the concept barrier with diagram in synchronous computation.
5.2. Explain Linear Barrier or Centralized Counter Implementation in Barrier.
5.3. Explain with diagram, barrier implementation in a message-passing system
5.4. Barrier Implementation using Tree construction.
5.5. Explain Barrier Implementation using Butterfly construction.
5.6. Explain the problem of deadlock in brief.
5.7. Explain prefix sub problem with example.

Unit 6. Load Balancing and Termination Detection [Approx. Marks 9/70]

6.1. What is load balancing? What are the issues we face if load balancing is not incorporated?
6.2. List various static load balancing algorithms.
6.3. What are the flaws/limitations of static load balancing?
6.4. Explain Centralized Dynamic Load Balancing with diagram
6.5. Explain Decentralized Dynamic Load Balancing with diagram
6.6. What is termination conditions?
6.7. Explain termination conditions using (a) Using Acknowledgement Messages.
6.8. Explain termination conditions using (b) Ring Termination Algorithms.
6.9. Explain termination conditions using (c) Tree Termination Algorithms.
6.10. Explain "Climbing a mountain" method to find shortest path algorithm

Unit 7. Programming with Shared Memory [Approx. Marks 10/70]

7.1. Draw a typical shared memory multiprocessor interconnection/architecture and explain the same in brief.
7.2. What are the options for programming multicomputer?
7.3. List some early parallel programming languages.
7.4. Explain the FORK-JOIN constructs to create concurrent processes, with neat diagram.
7.5. Differentiate between Processes and Threads.
7.6. Explain Thread create, join and detached with sample syntax.
7.7. What are the issues with accessing shared data? How the issues can be addressed with the idea of critical section and mutual exclusion?
7.8. Explain following concepts with respect to accessing shared data
(a) Locks (b) Deadlock (c) Semaphores (d) Monitor (e) Condition Variables
7.9. Explain OpenMP in detail.

Unit 8. Distributed Shared Memory Systems and Programming [Approx. Marks 10/70]

8.1. Explain Distributed shared memory
8.2. Explain the advantages/benefits of Distributed shared memory
8.3. Explain Central Server Algorithm to implement Distributed shared memory
8.4. Explain Migration Algorithm to implement Distributed shared memory
8.5. Explain Read Replication Algorithm to implement Distributed shared memory
8.6. Explain Full Replication Algorithm to implement Distributed shared memory