19CS8708 – MULTI-CORE ARCHITECTURES AND PROGRAMMING

Question Bank

2 – marks

1.What is the need for snooping protocol?

2.What are the issues available in handling the performance?

3.Differentiate single-core CPU and multi-core CPU.

4.How do you define the term false Sharing?

5.Analyze whether the complexity of an algorithm dominates run time.

6.What is meant by Semaphore?

7.Mention the types of wakeup calls in producer thread.

8.Write down the process of identifying the Synchronization.

9.Point out the effect of cancel construct.

10.What is the process of flush operation in OpenMP?

11.Write a mathematical formula for speedup and efficiency of parallel program.

12.What is meant by message queue?

Big Questions

1. (i) Briefly explain the concept of Single Instruction, Multiple Data System with neat diagram. (7)

(ii) Briefly explain Multiple Instruction, Multiple Data systems with neat diagram. (6)

2.Describe Symmetric Shared Memory architecture with neat diagram. (13)

3. (i)The interconnect plays a decisive role in the performance of both distributed- and shared-memory systems: even if the processors and memory have virtually unlimited performance, a slow interconnect will seriously degrade the overall performance of all but the simplest parallel program. How Shared-memory interconnects works? Justify with neat diagram. (7)

(ii) Write down the process of Distributed-memory interconnects for providing the Interconnection with neat diagram. (6)

4. (i) How Speedup and efficiency should affect the performance of the Multi-core processes. (4)

(ii) State Amdahl's law for Multi-core processor. (3)

(iii)Write down the process of analysing the Scalability in Multi-core processor. (3)

(iv) What is the process of Talking timings in a distributed-memory? (3)

5.(i) How communication between threads is achieved by Message queues for the requirement of communication between either the threads or the processes? (7)

(ii) Write down the process of communication within the threads using Named Pipes in detail. (6)

6.(i) Write down the concepts of Synchronization primitives for utilising the Spinlock (5)

(ii) How the Barriers are helpful for the situations where a number of threads have to all complete their work before any of the threads can start on the next task? (4)

(iii) Justify the process of Semaphores in the synchronization primitives. (4)

7 (i) Suppose two threads need to acquire mutex locks A and B to complete some task. If thread 1 has already acquired lock A and thread 2 has already acquired lock B, then A cannot make forward progress because it is waiting for lock B, and thread 2 cannot make progress because it is waiting for lock A. The two threads are deadlocked. How to solve this problem? (7)

(ii) A livelock traps threads in an unending loop releasing and acquiring locks. Livelocks can be caused by code to back out of deadlocks and explain the solution for Livelocks. (6)

8. Describe the ways to detect data races using various tools. (13)

9. Elaborate OpenMP Execution Model with a sample program. (13)

10. In a shared memory multiprocessor with a separate cache memory for each processor, it is possible to have many copies of any one instruction operand: one copy in the main memory and one in each cache memory. When one copy of an operand is changed, the other copies of the operand must be changed also. Cache coherence is the discipline that ensures that changes in the values of shared operands are propagated throughout the system in a timely fashion. How can you implement Cache Coherence? (15)

11. The shared memory programs will almost always have critical sections, which will require that we use some mutual exclusion mechanism such as a mutex. The calls to the mutex functions are overhead that‘s not present in the serial program, and the use of the mutex forces the parallel program to serialize execution of the critical section. Describe the performance issues of Multi-core Processors. (15)

12. (i) Briefly demonstrate the process of cache coherence through Snooping Cache coherence with neat diagram. (7)

(ii) Briefly explain about the Directory based Cache coherence and false sharing. (6)