CE1-R3: ADVANCED COMPUTER ARCHITECTURE

NOTE:

1. **Answer question 1 and any FOUR questions from 2 to 7.**
2. **Parts of the same question should be answered together and in the same sequence.**

Time: 3 Hours **Total Marks: 100**

1.

a) Comment on the guidelines to convert any blocking multi-stage interconnection network to its equivalent non-blocking counterpart.

b) Briefly comment on the major reasons for cache coherence problem in any multiprocessor system?

c) Define diameter with respect to static networks.

d) What are the characteristics of SIMD architecture?

e) Comment on the number and type of process states of any message passing system in a multiprocessor system.

f) What factors determine the performance of vector processors?

g) Estimate the effect of branch instructions on the performance of pipelined architectures with suitable set of parameters.

(7x4)

2.

a) Assume that an unpipelined machine has 10-ns clock cycle and it uses 4 cycles each for ALU operations and branch instructions and 5 cycles for memory operations. Assume that the relative frequencies of these operations are 40%, 20% and 40% respectively. Suppose that due to clock skew and setup, pipelining the machine adds 1 ns overhead to the clock. Ignoring any latency impact, how much speedup in the instruction execution rate will be gained from a pipeline?

b) From a given reservation table of a pipelined architecture (uni-function), how will find the minimal and maximal values of average latencies? Justify your answer

(6+12)

3. What is meant by cache-coherency? Explain with the help of a suitable example. State any three techniques to reduce cache miss. How does two-level cache increase performance? Derive the formula for average access time in a three-level cache?

([3+2+6+4+3])

4.

a) What is a vector processor? What are the properties of vector instructions? How are the two important issues like Vector Length and Stride tackled?

b) Vectorizing compilers generally detect loops that can be executed on a pipelined vector computer. Are the vectorization algorithms used by vectorizing compilers suitable for MIMD machine parallelization? Justify your answer.

(12+6)

5.

a) Explain the different types of dependences among instructions with suitable examples.

b) Consider the following loop:

```
for (i = 1; i <= 500; i++) {
          A[i] = A[i] +B[i];          /* Statement 1 or S1 */
          B[i+1] = C[i] +D[i];        /* Statement 2 or S2 */
          }
```

What are the dependences between S1 and S2? Is this loop parallel? If not, show how to make it parallel.

(10+8)

6.

a) How is a block found if it is in cache? What are the different write policies for cache?

b) Assume the following miss rates:

Size	Instruction Cache	Data Cache	Unified Cache
16 KB	0.64%	6.47%	2.87%
32 KB	0.39%	4.82%	1.99%

Which has the lower miss rate: a 16-KB instruction cache with a 16-KB data cache or a 32-KB unified cache? Assume a hit takes 1 clock cycle and the miss penalty is 50 clock cycles, and a load or store hit takes 1 extra clock cycle on a unified cache. What type of hazard the unified cache poses? What is the average memory access time in each case? Assume write-through caches with write buffer and ignore the difficulties due to hazard.

(10+8)

7.

a) What are dynamic networks? What are the characteristics of access time in such networks? What is Multistage Interconnection Network?

b) What do you mean by hazard? What are different types of hazards? What are the alternative techniques to reduce the data hazard?

(9+9)