Question 1: [CLO # 1] **[0.5 x 8 = 4 Points]**

Answer the following questions as True and False. No marks awarded without one sentence explanation.

a) Different types of instructions such as parallel, branch, pipelined can be issued for in-order execution.
False. Branch and pipelined issued in order. Parallel issues on one superscalar or two processors.

b) Memory system performance is captured by one parameter that is latency.
False. Latency + Bandwidth

c) Multithreading is one way to implement latency hiding.
True. Keeps memory unit busy with a queue of pending memory requests.

d) Fine granularity means more computation tasks and less communication.
False. Coarse grain is more computation and less communication.

e) Increasing memory capacity also increases the bandwidth.
False. Bandwidth is based on the architecture. Capacity can be increase by putting in more memory in empty slots.

f) A 40K size array can be stored completely in a cache having 32K cache lines, where each line contains 8 words.
32K x 8 = 256K words. 40K words or array can easy fit in this capacity.

g) Many shared memory systems can be connected together to form a distributed memory system.
True. For example a cluster of PCs

h) 12 FOLOPs is the term use to show 10^{12} floating-point operations per second by a processor.
False. FLOPs is the valid acronym. Tera FLOPs

Question 2: [CLO # 1 & 3] **[0.25 x 8 = 2 Points]**

Fill in the blanks. Write **one technical term** in your answer sheet.

a) Memory bandwidth can be improved by increasing the **block size**.

b) In **Spatial** locality, nearby instructions to recently executed instruction are likely to be executed soon.

c) The number of tasks into which a problem is decomposed determines its **granularity**.

d) Decomposition into a large number of tasks results in **fine-grained** decomposition and that into a small number of tasks results in a **coarse-grained** decomposition.

e) The maximum degree of concurrency is the maximum number of tasks that can be executed **in parallel**.

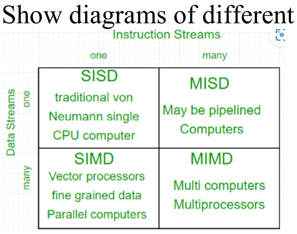
f) The length of the longest path in a task dependency graph is called the **critical path length.**

g) **Parallel overhead** can include factors such as: Task start-up time, Synchronisations, Data communications.

h) Maximum operations when multiplying a dense matrix with a vector are of the order of **n^3**

Question 3: [CLO # 1] **[4 Points]**

a) Show diagrams of different configurations of processing elements (PE) as proposed by Flynn. [0.5 x 4 = 2]



Instruction Streams

		one	many
Data Streams	one	SISD traditional von Neumann single CPU computer	MISD May be pipelined Computers
	many	SIMD Vector processors fine grained data Parallel computers	MIMD Multi computers Multiprocessors

b) Label each one suitable for data parallelism, task parallelism or both with short 1-2 sentence justification. [0.5 x 4 = 2]
SISD: None or implicit parallelism, **MISD**: Task parallelism, **SIMD**: Data Parallelism, **MIMD**: Both.

Question 4: [CLO # 3] **[4 Points]**

Assume that an n x n matrix **A** is multiplied by a vector (of length n) **b** to produce a vector **c**. Now answer the following:

a) Explain fine-grain and coarse-grain tasks creation for this multiplication that can run independently. [0.75 + 0.75]
Fine-grain tasks can be one row of matrix A multiplied by corresponding element of b to compute one value of c. Therefore, there could be n tasks at max. In case of coarse-grain, we can assign many row of matrix A to a single task. Both type of task can run independently.

b) **Which** task creation strategy in part (a) will perform better than the other will and **why**? Explain. [1+1]

- For smaller values of n both will work the same.
- For large values of n (~n = 1000) we need to think about limiting data accessing from memory and using cache line size. a) Fine-grain for n approaching 1000 will incur parallel overhead thread scheduling cost.
  b) Course-gain will perform better on multiple-core machines as different parts of data (matrix and vector) will sits in different cores until the end of the task.

c) Why do you think the cache inside the processor will speed-up this computation? [0.5]

- Pre-fetching data using cache lines – spatial locality.
- Large data caches will help reuse values, which are needed repeatedly – temporal locality.

Question 5: [CLO # 3] **[4 Points]**

Suppose we query the FLEX database with the where clause containing: Discipline = 'BSCS', Semester = 7, CGPA > 2.7, Failed courses <= 2, and HSC percentage > 75. Assume that each clause will generate a separate intermediate table of entries. Now:

a) Illustrate decomposition of the above query as a task dependency graph **with minimum critical path length**. [2]
Each clause can be plotted with some assumed dependency so each student must assume the dependency by them self and clearly specify critical path by (again) assuming weights to tasks. Copying textbook example be given ZERO.

b) Calculate maximum and average degree of concurrency using graph of part (a). Show **detail working** to get full marks. [2]
Separate working for maximum and average degree with detailed working. Write single figure values without explaining be given ZERO.

----------(O)----------