

1. (11) **True or False:**

- (1) Measuring performance on multiprocessors using linear speedup instead of execution time is a bad idea.
- (1) Amdahl's law applies to parallel computers.
- (1) You can predict cache performance of Program A by analyzing Program B.
- (1) The SPARC architecture uses condition codes.
- (1) It is not necessary to simulate very many instructions in order to get an accurate performance measure of the memory hierarchy.
- (1) Pipelining ideas can be implemented independent of technology
- (1) A program's locality behavior is constant over the run of an entire program.
- (1) Communication is not a significant problem for parallel processor systems.
- (1) Throughput and response time are the same thing.
- (1) The design of an instruction set has little impact on how pipelinable it is.
- (1) Professor Farrens never wore shorts to class.

2. (12) What do the following acronyms stand for?

SMP

DSM

COMA

CMP

SMT

TLB

UMA

CC-NUMA

MPP

MIMD

SIMD

SISD

**For the next 5 problems, circle the correct answer.**

3. (1) Which type of cache miss can be changed by altering the mapping scheme?

Capacity      Coherence      Compulsory      Conflict

4. (1) Which type of cache miss can be reduced by using longer lines?

Capacity      Coherence      Compulsory      Conflict

5. (1) Which type of operation is necessary in order to support synchronization?

Nuclear                  Atomic                  Radioactive

6. (1) Relaxing the requirement that Writes complete before Reads yields a model known as

Total Store Ordering      Partial Store Ordering      Weak Ordering

7. (2) To achieve a speedup of 80 using 100 processors, what percentage of the program must be parallelizable?

99.75%                  .9975%                  2.5%                  .25%                  5%

8. (11) Which of the following techniques would reduce the Miss Rate? Which would reduce the Hit Time? (Write "MR" to the left of the ones that would reduce the Miss Rate, and write "HT" to the right of the ones that reduce Hit Time.)

Small and simple caches

Larger block size

Bigger caches

Way prediction

Trace caches

Higher associativity

Multilevel caches

Pipelined caches

Non-blocking caches

Multibanked caches

Compiler optimizations

Victim cache

Priority to Read Misses

Critical Word First/Early Restart

Merging Write Buffer

Avoiding Address Translation when Indexing Cache

Hardware Prefetching

Software prefetching

**Very Short Answer:**

9. (4) If you are mainly worried about program size, what type of instruction set would you use? What if performance was your primary concern?

10. (2) Which is more effective, dynamic or static branch prediction?

11. (1) Do benchmarks remain valid indefinitely?

12. (4) Issuing multiple instructions per cycle puts tremendous pressure on what two parts of the machine?

13. (3) What is Amdahl's law (in words)?

14. (4) What is the primary difference between Scoreboarding and Tomasulo's algorithm?

**Short Answers:**

15. (3) What is the goal of the memory hierarchy? What two principles make it work?

16. (4) What is a victim cache, and how does it work?

17. (3) Branch predictors don't work well on a certain kind of control flow change (jump) that occurs very frequently. What is the jump in question, and what is one popular way to predict it?

18. (4) Why are there multiple dies per silicon wafer? Why not just fabricate one huge die per wafer?

19. (3) Write down the 3-term CPU performance equation developed in class.

**Slightly Longer Answers:**

20. (10) What is Cache Coherence, and why is it necessary? Snooping is one main approach to providing coherence - state what the other main approach is, and briefly outline how each of them work.

21. (10) Describe the difference between shared memory and message passing machines. Include the impact on design, cost, and programming model.

22. (5) What pair of instructions are used to implement a lock in RISC systems (as described in the text)? Describe how this pair works together in order to accomplish the goal.

23. (4) What is the definition of a basic block? Why is there a desire to create larger ones?

24. (6) Why is branch prediction important? What performance enhancing techniques have made it so? List 3 examples of existing Branch Prediction strategies in order of (average) increasing effectiveness.

25. (6) Explain pipelining - why is it so popular? What are the advantages? What are the disadvantages?

26. (6) Supporting precise interrupts in machines that allow out of order completion is a challenge. Briefly explain why.

27. (5) What is the maximum number of interrupts that could occur during a single clock cycle in a MIPS 5-stage pipeline? What are they?

28. (5) Assume a relatively large fully associative write-back cache that contains no valid data. Given the following sequence of 5 memory operations (the address of the operation is in the square brackets):

WriteMem[100]

ReadMem[100]

WriteMem[100]

WriteMem[200]

WriteMem[100]

What are the number of hits and misses when using write allocate versus no-write allocate?

29. (10) Suppose that in 1000 memory references there are 40 misses in the first-level cache and 20 misses in the second-level cache. What are the various miss rates? In addition, assume the miss penalty for the L2 cache is 200 clock cycles, the hit time of the L2 cache is 10 clock cycles, the hit time of the L1 cache is 1 clock cycle, and there are 1.5 memory references per instruction. What is the average memory access time in cycles? What is the average stall cycles per instruction? (Ignore writes, and assume a base CPI of 1.)



30. (8) In class, we talked about the cycle by cycle steps that occur on different interrupts. For example, here is what happens if there is an illegal operand interrupt generated by instruction i+1:

	1	2	3	4	5	6	7	8	9
i	IF	ID	EX	MEM	WB				
i+1		IF	ID	EX	MEM	WB	<- Interrupt detected		
i+2			IF	ID	EX	MEM	WB	<- Instruction Squashed	
i+3				IF	ID	EX	MEM	WB	<- Trap Handler fetched
i+4					IF	ID	EX	MEM	WB

---

Fill out the following table if instruction i+1 experiences a fault in the Mem stage (page fault, for example):

	1	2	3	4	5	6	7	8	9	10
i	IF	ID	EX	MEM	WB					
i+1		IF	ID	EX	MEM	WB				
i+2			IF	ID	EX	MEM	WB			
i+3				IF	ID	EX	MEM	WB		
i+4					IF	ID	EX	MEM	WB	
i+5						IF	ID	EX	MEM	WB

---

What happens in this case?

	1	2	3	4	5	6	7	8	9	10
i	IF	ID	EX	MEM	WB	<- Data write causes Page Fault				
i+1		IF	ID	EX	MEM	WB	<- Illegal Opcode			
i+2			IF	ID	EX	MEM	WB			
i+3				IF	ID	EX	MEM	WB		
i+4					IF	ID	EX	MEM	WB	
i+5						IF	ID	EX	MEM	WB