

Name: _____

Rules and Hints

- You may use one handwritten 8.5×11 " cheat sheet (front and back). This is the only additional resource you may consult during this exam. *No calculators.*
- You may write your answers in the form $[mathematical\ expression][units]$. There is no need to actually do the arithmetic.
- Write your answers on scratch paper. Clearly label your answer to each question.

Grade

	Your Score	Max Score
<i>Problem 1: Cache tracing</i>		24
<i>Problem 2: Cache and page table sizing</i>		20
<i>Problem 3: Address translation</i>		12
<i>Problem 4: I/O</i>		34
<i>Problem 5: Cache coherence</i>		10
Total		100

Problem 1: Cache tracing (24 points)

A hilariously tiny memory system has 12-bit addresses. Consider the following sequence of memory references:

1. 1011 0101 1100
2. 1111 0101 1100
3. 1011 0101 1110
4. 1011 0111 1100
5. 1011 1101 1100
6. 1111 1111 1100

Part A (12 points)

If the L1 cache has 32B of data and is direct-mapped with 4-byte blocks, give the hit rate of this sequence and show the final state of the cache.

Part B (12 points)

If the L1 cache has 32B of data and is 2-way set-associative with 2-byte blocks and LRU replacement, give the hit rate of this sequence and show the final state of the cache.

Problem 2: Cache and page table sizing (20 points)

Part A: Cache sizing (10 points)

A 2-way set-associative, write-through cache has 32 KB of data in 32B blocks. How much metadata, in total, does this cache need? Assume 32-bit memory addresses.

Part B: Page table sizing (10 points)

A system has 32-bit virtual addresses and 38-bit physical addresses, with 8KB pages. Assuming a single reference bit and three permission bits per page, how much *total* space will each process need for a flat page table?

Problem 3: Address translation (12 points)

A hilariously tiny system has 8-bit virtual addresses and 10-bit physical addresses. The page size is 32B.

If each virtual page P for Process A is mapped to physical page $3P$, draw Process A's page table and translate virtual address 10011001 to its corresponding physical address.

Problem 4: I/O (34 points)

The disks used in this problem are 1 TB disks with an average seek time of 4 ms, a rotation speed of 7200 rpm, and a transfer speed of 100 MB/s.

Part A: Disk performance (10 points)

How long does a small access (say, 512B) take on one of these disks? You can abbreviate this number as S from this point forward.

How long does a 12 MB access take on *one* of these disks? You can abbreviate this number as B from this point forward.

Parts B–D

(24 points) You have exactly 12 of these disks and cannot buy more. Fill out the table below to indicate for each RAID level:

- Part B (6 points): How much non-redundant data (in TB) can you fit in a 12-disk array?
- Part C (9 points): How many small *writes* can you do at one time across the 12-disk array?
- Part D (9 points): In terms of your answers to Part A, how long does a single 12MB write take?

Assume 0.25 MB (256 KB) chunk sizes for RAID 5.

	RAID 0	RAID 1	RAID 5
B. Data in TB			
C. # small writes			
D. Large write time			

Problem 5: Cache coherence (10 points)

Consider a simpler write-back, write-invalidate coherence protocol that only has 2 states: Invalid or Valid. For your reference, the 3-state MSI diagram is below.

Part A: State diagram (7 points)

On the next page, draw a state diagram for your protocol. In each stage, you need to deal with the following possible inputs:

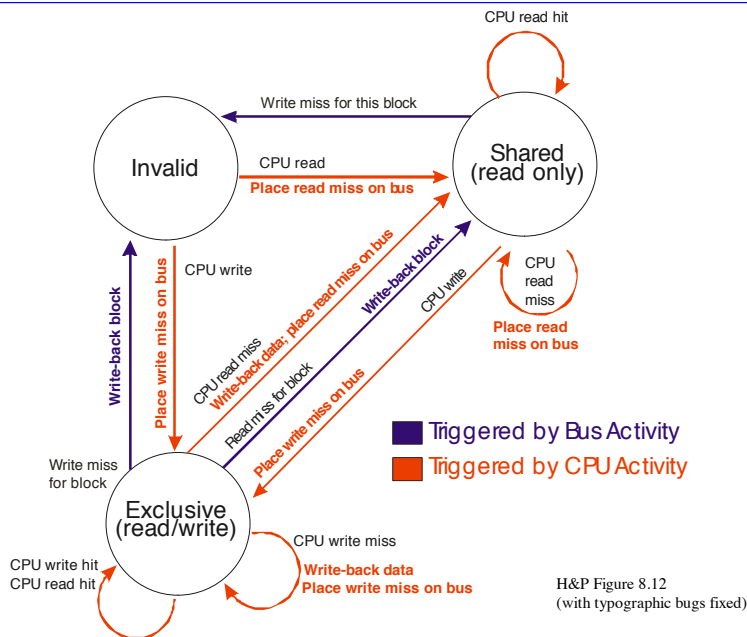
- PRH: Processor read hit
- PRM: Processor read miss
- PWH: Processor write hit
- PWM: Processor write miss
- BRM: Bus read miss (that is, "overheard" read miss from another processor)
- BWM: Bus write miss

For each *possible* input, show the state transition as well as any of the following actions:

- RM2b: put read miss on bus
- WM2b: put write miss on bus
- WB: write back data if dirty (assume you have a dirty bit).

Part B: 3 points

Below your diagram, state the disadvantage of this protocol compared to MSI. Be specific: what kind(s) of access patterns will perform worse?



[Diagram for Problem 5]