Name: _____Key_____ Uname: _____Key_____

You have 20 minutes for this quiz. The quiz is closed notes/closed book. If you should finish early, you are welcome to turn in your quiz and step out of the room until the lecture starts. **There are 2 pages to this quiz.**

I have neither given nor received aid on this quiz, nor observed anyone else doing so.

_____

---

1. Consider the following access pattern: A, B, C, B, A. Assume that A, B, and C are memory addresses each of which are in a different block of memory. Further, assume A, B and C are generated in a uniformly random way.
   a. What is the probability that the second instance of "B" will be a hit on an 8-line direct-mapped cache? Briefly show your work. **[10]**

      (7/8)=0.875

   b. What is the probability that the second instance of "A" will be a hit on a 4-line 2-way associative cache? Again, briefly show your work. **[15]**

      B and C have to go to the same set as A to get a miss. The odds of that are $(1/2)^2$. So the odds of a hit is ¾ or 0.75.

2. Consider a virtually-addressed cache that is used in a MESI-bus system. What additional issue would this type of cache create when snooping? Your answer must be 20 words or less. **[10]**

   You'll need to convert the bus's physical address to a virtual address (reverse TLB lookup).

3. Consider a processor which, when running a given application performs 2 billion loads and 1 billion stores per second. Assume the following is true:
   - The processor's multi-level cache system gets a 97% hit rate on both loads and stores.
   - Cache lines are 32-bytes in size
   - There is no prefetching and the instruction cache never misses.
   - The cache is ***write-thru*** and write-allocate.
   - There are no coherence misses.
   - All loads and stores are to 4-byte values.

   a. What is the read bandwidth (bytes/second) on the bus? Show your work. **[10]**

      $2*10^9$ loads * 0.03 transactions/load * 32 bytes/transaction +
      $1*10^9$ stores * 0.03 transactions/load * 32 bytes/transaction =
      2.88 billion bytes read per second.

   b. What is the write bandwidth (bytes/second) on the bus? Show your work. **[5]**

      All stores result in a 4-byte write to memory. So, 4 billion bytes written/second.

4. Consider a case of having 3 processors using a snoopy MESI protocol where the memories can snarf data. All three have a 2-line direct-mapped cache with ***each line consisting of 8 bytes***. The caches begin with all lines marked as invalid. Fill in the following tables indicating
   - If the processor gets a hit or a miss in its cache
   - If a HIT or HITM (or nothing) occurs on the bus during snoop.
   - What bus transaction(s) (if any) the processor performs (BRL, BWL, BRIL, BIL)
   - For misses only, indicate if the miss is compulsory, capacity, conflict, or coherence. A coherence miss is one where there would have been a hit, had some other processor not interfered.

Finally, indicate the state of the processor after all of these memory operations have completed. The operations occur in the order shown. **[50 points, -2 per wrong or blank, minimum of 0]**

Processor	Address	Read/ Write	Cache Hit/Miss	Bus transaction(s)	HIT/ HITM	State after transaction	"4C" miss type (if any)
1	0x100	Read	Miss	BRL	--	E	Compulsory
1	0x008	Write	Miss	BRIL	--	M	Compulsory
1	0x018	Read	Miss	BRL/BWL	--	E	Compulsory
1	0x100	Write	Hit	--	--	M	--
2	0x018	Read	Miss	BRL	HIT	S	Compulsory
2	0x100	Read	Miss	BRL	HITM	S	Compulsory
2	0x01F	Write	Miss	BIL	HIT	M	Coherence
1	0x018	Read	Miss	BRL	HITM	S	Coherence
3	0x01F	Write	Miss	BRIL	HIT	M	Compulsory
1	0x01F	Read	Miss	BRL	HITM	S	Coherence

Final state:

	Proc 1	
	Address	**State**
Set 0	**0x100**	**S**
Set 1	**0x018**	**S**

	Proc 2	
	Address	**State**
Set 0	**0x100**	**S**
Set 1	**0x018**	**I**

	Proc 3	
	Address	**State**
Set 0		
Set 1	**0x018**	S

Notes:
- In the final state we treat empty and invalid cache lines as equivalent.
- The answers only store the address matching the start of the cache line, but will accept any address in the correct line (Procs 1 / 3: 0x018 vs 0x01F)
- Bus transactions are those originating from the processor in each line, the effect of another processor responding with data to the requesting proc is handled by the HIT/HITM column.