Name: ______________________________ Uname: _____________________

You have 25 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 two different *four-line caches*. Assuming that the accesses "A", "B", and "C" are each to different cache lines and that the exact location of A, B and C are independent of each other, write the chance of each access being a hit. Assume both caches are initially invalid. You may provide your answers as fractions. [**10 points, -1 per wrong or blank answer]**

Reference	**A**	**B**	**C**	**B**	**A**	**A**	**B**	**C**	**A**
Hit chance on direct-mapped cache	0%	0%							
Hit chance on 2-way set-associative cache	0%	0%							

2) There are some significant limitations when it comes to having the compiler hoist loads. **[5 points each]**
   a) What issue are we primarily concerned about when we hoist a load above a store?

   b) What issue are we primarily concerned about when we hoist a load above a branch?

6. 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 16 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 caused an invalidation of that line.

   In the event more than one bus transaction occurs due to a given memory read/write indicate the response for each bus transaction. Finally, indicate the state of the processor after all of these memory operations have completed. The operations occur in the order shown. **[10 points, -0.5 per wrong or blank, minimum of 0]**

Processor	**Address**	**Read/Write**	**Bus transaction(s)**	**Hit/Miss**	**HIT/ HITM**	**"4C" miss type (if any)**
1	0x200	Read				
1	0x200	Write				
1	0x210	Read				
1	0x100	Read				
2	0x200	Read				
2	0x210	Write				
2	0x100	Write				
3	0x100	Read				
1	0x100	Write				
2	0x210	Read				

Final state:

	Proc 1	
	Address	**State**
Set 0		
Set 1		

	Proc 2	
	Address	**State**
Set 0		
Set 1		

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