EECS 570 Final Exam  
Winter 2012

Name: ____________________________________    unique name: _______________

Sign the honor code:

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

Scores:

<table>
<thead>
<tr>
<th>#</th>
<th>Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>/ 20</td>
</tr>
<tr>
<td>2</td>
<td>/ 20</td>
</tr>
<tr>
<td>3</td>
<td>/ 24</td>
</tr>
<tr>
<td>4</td>
<td>/ 18</td>
</tr>
<tr>
<td>5</td>
<td>/ 18</td>
</tr>
<tr>
<td>Total</td>
<td>/ 100</td>
</tr>
</tbody>
</table>

NOTES:
- Closed book, closed notes.
- Calculators are allowed, but no PDAs, Portables, Cell phones, etc.
- Don’t spend too much time on any one problem.
- You have about 90 minutes for the exam (avg. 18 minutes per problem).
- There are 8 pages in the exam (including this one). Please ensure you have all pages.
- Be sure to show work and explain what you’ve done when asked to do so.
1) Speculative Multithreading [20 points]

a) How does speculative multithreading differ from transactional memory? [6 points]

b) Describe two key roles the compiler plays in the Multiscalar design. [8 points]

c) What is the purpose of the address resolution buffer (ARB) in the Multiscalar system. [6 points]
2) **Ring Interconnects [20 points]**

a) What is the average latency in a unidirectional ring with N nodes? [4 points]

b) Suppose we make the ring bidirectional. How does this change your answers for part (a)? [4 points]

c) Consider an 8-node bidirectional ring that uses an oblivious routing protocol.

i) What is the worst-case traffic pattern for this topology? Provide the *permutation function* \( \text{dest} = \pi(\text{src}) \) for the traffic pattern. (A permutation function provides the destination node id for each message as a function of the source node id). [4 points]

ii) Describe an oblivious routing algorithm that achieves the best possible throughput for this traffic pattern. [4 points]

iii) What is that throughput (expressed as a fraction of the maximum link bandwidth)? [4 points]
3) Routing & Flow Control [24 points]

a) Can a network with virtual cut-through routing deadlock? Explain why or why not? [6 points]

b) State an advantage of on-off flow control relative to credit-based flow control. [6 points]
c) Consider a 16-node 2D torus topology that uses minimal deterministic routing. Assume the routers are optimized with lookahead routing, speculative virtual channel allocation, and bypassing when input buffers are empty. Further assume that the link between routers is pipelined and takes 2 cycles to traverse. What is the longest a flit can take to traverse the network if it is injected by itself (i.e., it encounters no congestion)? Be sure to include time spent in the router at the ingress and egress nodes (i.e., a flit starts in the first stage of the router at the ingress node, and arrives in the cycle after switch traversal at the egress node). [6 points]

d) State an advantage and a disadvantage of deflection (a.k.a hot potato) routing. [6 points]
4) GPUs [18 points]

a) Why is a branch that is 95% taken and 5% not taken problematic under the SIMT threading model of GPUs? [6 points]

b) How does a GPU hide the long latency of accesses to global memory? [6 points]

c) Briefly explain why CUDA code optimized for one GPU might run inefficiently when executed on a newer model of GPU. [6 points]
5) Memory Consistency [18 points]

a) Suppose the following loads and stores are executed on a system that implements the Release Consistency (RC_sc) memory model.

   Load X
   (1)
   Load.acquire Lock
   (2)
   Store Y
   (3)
   Store.release Lock
   (4)
   Store Z

For each of the three accesses to X, Y, and Z, list all of the numbered points in the execution where those operations can legally be executed. (In other words, indicate if it is legal to re-order each of those three instructions such that they occur at each of the numbered points relative to the other operations). [4 points]

Load X:

Store Y:

Store Z:

b) Briefly explain why store atomicity is difficult to maintain in update-based coherence protocols. [4 points]
c) Describe the design of a store buffer for a processor that implements a relaxed memory consistency model like Sun’s RMO (i.e., it uses fence instructions to enforce ordering). Draw a picture of your design. Be sure to state (i) under what circumstances a store instruction may drain from the store buffer into the cache hierarchy, and (ii) how fence instructions are implemented. [10 points]