CSE 471 Autumn 1998

Computer Design and Organization

Final Exam

Friday December 11th

NAME : ______________________________

Do all your work on these pages. Do not add any pages. Use back pages if necessary. Show your work to get partial credit.

This exam is worth 100 points. After each question, you will find the number of points it is worth. You should spend approximately x minutes on a question worth x points (e.g., 25 minutes on question 1 worth 25 points). That will leave you with 20 minutes to look over your work.

I. 25 points ____________

II. 10 points ____________

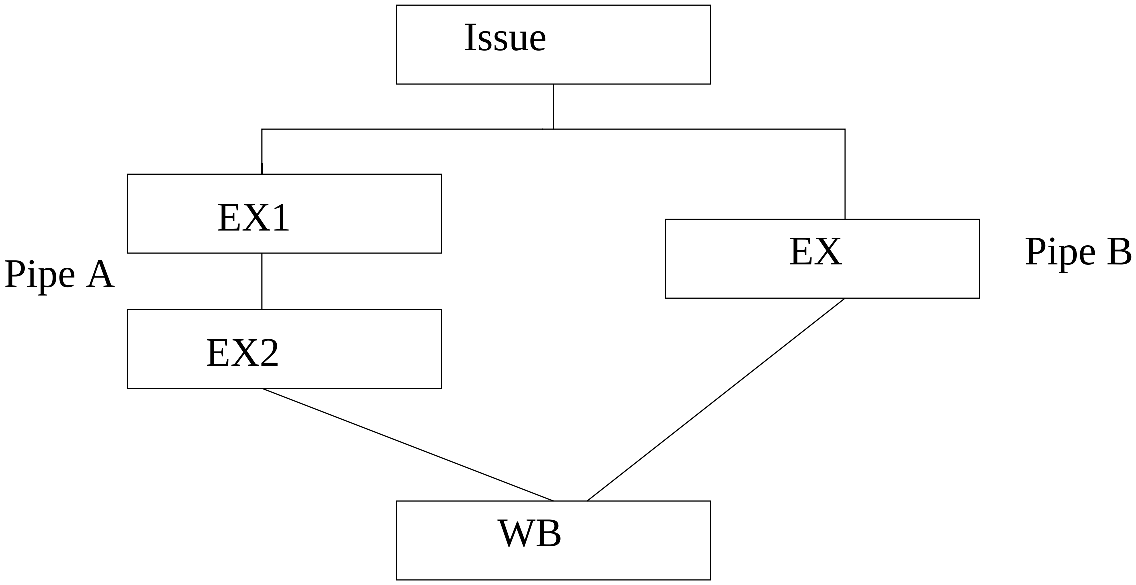
III. 20 points ____________

IV. 20 points ____________

V. 25 points ____________

I. (25 points) (This question continues on the next page)

Consider the following simple dual-issue CPU (cf. Figure below) for arithmetic-logical operations only. It has 2 pipelined functional units A and B that share an Instruction issue stage (fetch + decode) and a *single* Register Write-back stage. Allocation of the Register Write-back stage is done during the Instruction issue stage. Forwarding is implemented, i.e., the execute latency of A is 1 (result available at the output of EX2) and that of B is 0. Instructions are required to execute either on A or on B (this is dictated by their opcodes, i.e., there is no choice) and they have to be issued *in-order* but do not necessarily complete in order.



1. At a given time t is it possible to have instructions in progress simultaneously (if no, justify your answer in one sentence; if yes, just say yes)

   - in stages EX1 (pipe A), EX2 (pipe A), and EX (pipe B).

   - in stages EX2 (pipe A) and EX (pipe B).

   - in stages EX1 (pipe A) and EX (pipe B).

   - in stages EX1 (pipe A) and EX2 (pipe A)

2. At time t, two instructions, say i and $i+1$, are in the issue stage and the EX1, EX2, and EX stages may or may not be processing previous instructions. Instruction i is to execute in pipe A and instruction $i+1$ in pipe B. State the conditions, i.e., occupancy of pipe stages by previous instructions and data hazards, under which:

   (a) Both instructions will be in their execute stage, respectively EX1 (pipe A) and EX (pipe B) at time $t+1$

   (b) No instruction will be in its execute stage at time $t+1$

3. What are the changes in the conditions for (a) and (b) above (if any) when now instruction i has to execute on pipe B and instruction $i+1$ on pipe A.

II. (10 points)

1. What are the 3 fields in the bit-string representation of a floating-point number?

2. Within that representation, what is a normalized number?

3. Array A is an array of positive normalized floating-point single-precision (32 bits) elements. The array is to be sorted in ascending order. Can one use an algorithm which considers each 32-bit element as a 32-bit integer? Justify your answer.

III. (20 points)

Consider the following paragraph contained in the description of a recent product, a single-issue pipelined processor:
"The memory system comprises an 8KB data cache, 2-way set-associative, with 4 32-bit words per line. It is write-through and does not allocate on a write miss. Line fetch is performed with the addressed word first and is non-blocking, allowing hit-under-miss."

What do you think happens:

- On a cache read hit
- On a cache read miss (in particular what is the implication of "addressed word first" and does the pipeline always stall to wait for the miss to be resolved, i.e., how do you interpret "non-blocking, allowing hit-under-miss")
- On a cache write hit
- On a cache write miss

IV. (20 points) (This question continues on the next page)

1. What is the average memory access time (for data) of a memory hierarchy that consists of

   - A direct-mapped L1 cache with hit rate of 90% and access time of 1 cycle (on a hit). Read and write misses will incur the same penalty.
   - A 4-way set associative L2 cache with hit rate of 60% and access time of 10 cycles (on a hit)
   - The access time to main memory is 100 cycles

2. A victim cache of 4 entries is now added to the memory hierarchy as an assist to the L1 cache. Explain what happens on:

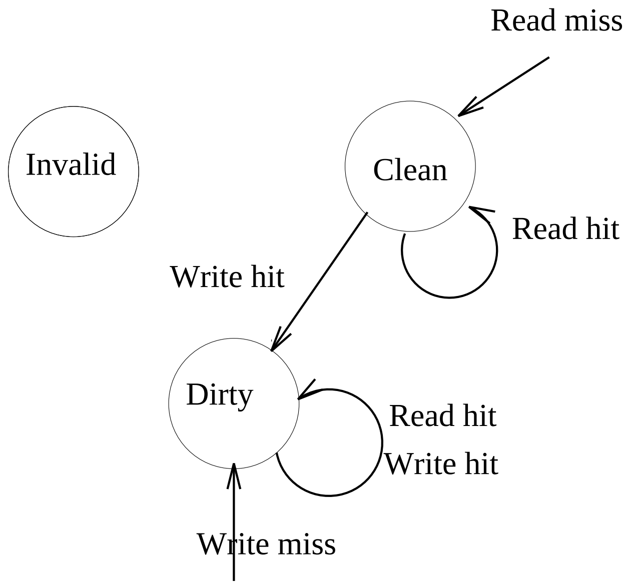
   - An L1 read hit
   - An L1 read miss

3. Assume that 25% of the misses in L1 hit in the victim cache and that the access time (on a hit) to the victim cache is 2 cycles. Hit rates to L1 and L2 remain the same. What is the new average memory access time.

V. (25 points) (This question continues on the next page)

1. The state diagram below shows the 3 necessary states for a write-invalidate snoopy cache coherence protocol as well as the cache state transitions based on requests from the CPU to which the cache is attached. Complete the diagram with transitions based on requests snooped from the bus.

   Explain what happens on read/write hits and misses.



2. Assume now that 3 processors and associated write-back, write allocate direct-mapped caches $C1, C2$ and $C3$ are attached to the bus. Addresses A and B belong to different lines but map to the same entries in the caches. Assume further that it takes much less than 100 cycles to complete a miss and associated transactions on the bus. Initially the cache entries corresponding to A (and B) are Invalid in all 3 caches. Show the actions that are taken and the states of the entries in the caches when the following read and write instructions are executed by the various processors under the cache coherency protocol of the previous page.

Time	Processor-Cache 1	Processor-Cache 2	Processor-Cache 3
0	Read A		
100		Read A	
200			Read B
300	Write A		
400		Read B	
500	Read B		
600			Read A