EECS 370 Final Exam

Winter 2002, Prof. Mark Brehob

Name: _____ UM ID: _____

Unique Name (print) _____

Sign the honor code:

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

_____

Scores:

#	Points
1	/50
2.1	/10
2.2	/5
2.3	/15
2.4	/15
2.5	/5
Total	**/100**

NOTES:

- Closed book and closed notes
- Non-programmable calculators only
- Don't spend too much time on any one problem.
- You have 120 minutes for the exam.
- Multiple-choice questions and certain other questions will be graded as right or wrong with no partial credit given.
- **The last problem is much harder than its point value. We suggest you not work on it until you have the rest of the exam completed.**

Section 1: Short answer and multiple choice. (52 points)

1. **Answer 3 of the following 4 questions.** *Cross out* the one you don't wish to have graded. If we can't tell which 3 you want graded, we will grade the first 3. (12 points, 4 each)

    a) Define the term "Amdahl's law"

    b) What is a stack frame?

    c) What is a structural hazard?

    d) What is an inverted page table?

Multiple choice (24 points, 4 each)

A computer system called the HOB2000 has the following properties:

- 5-stage pipeline as in project 3.
- L1 cache: 8KB 2-way associative; unified instruction and data; virtually indexed and physically tagged; 32-byte blocks; Access time is 1 cycle. (does not cause a stall on a hit!)
- TLB: 4 entry, fully associative; Access time is 1 cycle (also does not cause a stall.)
- L2 cache: 256KB 16-way associative; 32-byte blocks; exclusive of the L1; also unified instruction and data.
- Memory: 512MB, 100ns access time.
- Disk: 40GB. Rotates at 10,000RPM. Seek time 5ms, transfer rate, 50MB/sec, other overhead 3ms.
- Misc: Page size is 4KB. Byte addressable. 32-bit address space. Page table entries are 16 bytes. Processor speed is 1GHz.

2. How many bits are used to index the L1 cache? The L2 cache?
   a) 7 index bits for the L1, 9 index bits for the L2
   b) 6 index bits for the L1, 12 index bits for the L2
   c) 7 index bits for the L1, 11 index bits for the L2
   d) 6 index bits for the L1, 13 index bits for the L2
   e) none of the above are correct.

3. What is the size of the page table?
   a) 4MB
   b) 8MB
   c) 16MB
   d) 32MB
   e) 64MB

4. If the page table register (PTR) holds the value 0xFF00, what is the address of the page table entry (PTE) that would be accessed to translate the virtual address 0x0010010C?
   a) 0x10000
   b) 0xFF10
   c) 0x1FF00
   d) 0x10F00
   e) none of the above

5. Which of the following are TRUE of the HOB2000?
      i. If the TLB access time were changed to 2 cycles, the L1 access time would also *effectively* go up to 2 cycles.
      ii. The L2 cache is almost certainly virtually addressed
      iii. It takes about 100ns to do a DRAM memory access.
   a) i only
   b) ii only
   c) ii and iii only
   d) i and iii only
   e) None of i, ii, or iii is true.

The CRUMMY benchmark has the following properties:

- 50% of all instructions are adds
- 10% of all instructions are sw
- 20% of all instructions are lw
- 20% of all instructions are beq
- 100% hit rate in the L1 and TLB.
- 40% of all branches are not-taken
- 50% of loads are followed by a dependent instruction

6. What is the expected CPI of the CRUMMY benchmark on the HOB2000? Choose the closest answer.
   a) 1.1
   b) 1.2
   c) 1.3
   d) 1.4
   e) 1.5

7. If the CRUMMY benchmark had a 90% hit rate in the TLB what would be the CPI on the HOB2000? Assume the L1 still gets a 100% hit rate and that the L2 does not hold any page table entries. Choose the closest answer:
   a) 2
   b) 4
   c) 8
   d) 16
   e) 32

More short answer

8. Consider three different 4-entry fully associative caches in a **word** addressable machine with 1-word blocks. The three caches are identical other than replacement policy. One uses FIFO another LRU and the other OPT (optimal) (14)

a) Consider the following access pattern: 1, 2, 5, 6, 1, 3, 1, 2, 1, 6. What is the hit rate for each of the 3 caches? (6)

FIFO hit rate: _____ LRU hit rate: _____ OPT hit rate: _____

b) Provide the shortest access pattern you can where the FIFO cache will outperform the LRU cache. (4)

c) Provide the shortest access pattern you can where the LRU cache will outperform the FIFO cache. (4)

Section II (50 points)

1. The 3 C's (10 points)

Consider the following memory reference pattern to word addresses:

2, 4, 6, 8, 10, 10, 8, 6, 4, 2

Assume you have a direct-mapped cache with Z blocks. Each of those blocks are 1 word in size. Now answer the following questions. It may be that no value of Z will satisfy one or more of these questions. In that case write "There are no such values."

a) For what values of Z will you get no conflict misses?

b) For what values of Z will you get no capacity misses?

c) For what values of Z will you get no compulsory misses?

2. Caching problem (5)

What is the maximum possible size of a virtually-indexed, physically-tagged cache given a page size of 8KB, a cache block size of 32 bytes, and restricting the cache to be 2-way set associative? Briefly explain your answer (5)

3. Virtual Memory (15 points)

Consider a system that has:

- A 16-byte cache line
- 64 byte, 2-way set-associative cache
- A 4 KB page size
- A *direct-mapped* TLB with two entries.
- A physical memory of 16KB
- **A page table which is stored in physical page 0 and can never be evicted**
- A 20-bit, byte addressable, virtual-address space.

The page table initially has virtual page 0 in physical page 1, virtual page 1 in physical page 2, and no valid data in the other physical page(s). The TLB holds information on both virtual pages 0 and 1. Assume that the OS manages the TLB and that the OS puts the most recent page referenced into the TLB on a TLB miss. Further the page table is managed by a fully-associative LRU policy. Virtual page 1 is the least recently used page.

The following addresses are referenced sequentially. For each reference indicate:

- The Virtual page number (in hex)
- TLB miss? (Y or N)
- Page fault? (Y or N)
- Physical address translated to.

Do ***not*** worry about recording accesses to the page table itself.

Virtual address	**Virtual Page number (in hex)**	**TLB miss?**	**Page fault?**	**Physical address (after page fault if any)**
0x00F0C				
0x01F0C				
0x20F0C				
0x00100				
0x00200				
0x30000				
0x01FFF				
0x00200				

4. Pipelines and multi-cycle computers (15)

Consider an LC2K2 benchmark with the following distribution of instructions:

Lw	25%		Sw	15%
Add	20%		Nand	20%
Beq	20%			

- Branches are taken 60% of the time.
- A load word is immediately followed by an instruction dependent on the result of the load word 20% of the time.
- The benchmark will execute billions of instructions.

Now consider two implementations of the LC2K2 ISA. The first is the pipelined implementation from project 3. The second is a multi-cycle, unpipelined implementation. It takes 3 cycles on loads and stores and 2 cycles on all other instructions.

If the pipelined implementation runs at 200MHz, at what frequency must the multi-cycle implementation run in order to complete the benchmark at the same time as the pipelined implementation? ***You must clearly show your work for credit!*** You may ignore impacts of memory latency (assume all memory references are resolved in 1 clock).

4. Hard Question(5)

Answer 1 of the following 2 questions. *No partial credit will be given on this question.* The work required for this question can be large compared to its point value, so we recommend you do this problem last.

a) You need to maintain LRU ordering for an N-way set-associative cache. You will do this using a Y-bit field to keep track of the order. You may not use anything other than these Y bits to keep track of order. *Exactly* how many bits do you need in terms of *N*?

b) The following C program is run (with no optimizations) on a MIPS machine with a cache that has four-word (16-byte) blocks, holds 256 bytes of data and is direct-mapped:

```
int I,j,c,stride,array[256];
...
for (I=0;I<10000;I++)
{
      j=j+stride%256;
      c+=array[j];
}
```

If we consider only the cache activity generated by references to the array and we assume that integers are words, what is the expected miss rate when the cache is direct-mapped and the stride is 4? If the stride is 64? 63? Provide an answer that is correct to two significant digits for all three values of stride (4, 64, 63).