Caches and Virtual Memory

Assigned Wednesday, August 6, 2014
Suggested Completion Date *Wednesday, August 13, 2014*

Introduction

The purpose of written homework assignments is to get you thinking about the topics being covered in lecture and in readings in the textbook which are not represented in the hands-on, programming lab assignments. These written assignments also better prepare you for course examinations. It is worth noting that the book contains many practice problems similar to the problems we ask on these written assignments. The solutions for those practice problems are located at the end of each chapter and should give you a feel for the kind of answers we expect you to turn in for these kind of assignments.

Logistics

These written homeworks will not turned in for credit like the programming lab assignments. There also won't be solutions provided, but the forums are open to all discussion about the homework. We encourage you to complete the homework to the best of your ability and then discuss your findings/questions with your peers on the forums.

Questions

Answer the following problems:

1. Homework Problem 6.37 (pg 636)
   This problem tests your ability to predict the cache behavior of C code. You are given the following code to analyze:

```
1     int x[2][128];
2     int i;
3     int sum = 0;
4
5     for (i = 0; i < 128; i++) {
6        sum += x[0][i] * x[1][i];
7     }
```

   Assume we execute this under the following conditions:

   - `sizeof(int) = 4`

- Array `x` begins at memory address `0x0` and is stored in row-major order.
- In each case below, the cache is initially empty.
- The only memory accesses are to the entries of the array `x`

Given these assumptions, estimate the miss rates for the following cases:

A. Case 1: Assume the cache is 512 bytes, direct-mapped, with 16-byte cache blocks. What is the miss rate?
B. Case 2: What is the miss rate if we double the cache size to 1024 bytes?
C. Case 3: Now assume the cache is 512 bytes, two-way set associative using an LRU replacement policy, with 16-byte cache blocks. What is the cache miss rate?
D. For Case 2, will a larger cache size help to reduce the miss rate? Why or why not?
E. For Case 3, will a larger block size help to reduce the miss rate? Why or why not?

2. The following program may have different outputs depending on the order that processes run in; list *all* of the possible outputs. Assume that the `fork()` call cannot fail.

```
1     int main() {
2        int x = 3;
3
4        if (fork() != 0) {
5           x++;
6           printf("x=%d\n", x);
7        }
8
9        x--;
10       printf("x=%d\n", x);
11       exit(0);
12    }
```

3. Practice Problem 9.2 (pg 781)
Determine the number of page table entries (PTEs) that are needed for the following combinations of virtual address size (n) and page size (P):

n	$P=2^p$	No. PTEs
16	4K	_____
16	8K	_____
32	4K	_____
32	8K	_____
64	4K*	_____
64	4096K**	_____

* An x86-64 system with standard pages
** An x86-64 system with huge pages

4. Practice Problem 9.3 (pg 790)
Given a 32-bit virtual address space and a 24-bit physical address, determine the number of bits in the VPN, VPO, PPN, and PPO for the following page sizes P:

P	No. VPN bits	No. VPO bits	No. PPN bits	No. PPO bits
1 KB	_____	_____	_____	_____
2 KB	_____	_____	_____	_____
4 KB	_____	_____	_____	_____
8 KB	_____	_____	_____	_____

Also, consider a 64-bit virtual address space and a 36-bit physical address; repeat this problem for P = 4 KB and P = 4096 KB.

5. Homework Problem 9.11 (pg 849)
   In the following series of problems, you are to show how the example memory system in Section 9.6.4 translates a virtual address into a physical address and accesses the cache. For the given virtual address, indicate the TLB entry accessed, the physical address, and the cache byte value returned. Indicate whether the TLB misses, whether a page fault occurs, and whether a cache miss occurs. If there is a cache miss, enter "-" for "Cache Byte returned". If there is a page fault, enter "-" for "PPN" and leave parts C and D blank.

   The book's solution to practice problem 9.4 could be very helpful to understand before working on these problems.

   Virtual address: `0x027c`

   A. Virtual address format:

13	12	11	10	9	8	7	6	5	4	3	2	1	0

   B. Address translation:

Parameter	Value
VPN	_____
TLB index	_____
TLB tag	_____
TLB hit? (Y/N)	_____
Page fault? (Y/N)	_____
PPN	_____

   C. Physical address format:

11	10	9	8	7	6	5	4	3	2	1	0

   D. Physical memory reference:

Parameter	Value
Byte offset	_____
Cache index	_____
Cache tag	_____
Cache hit? (Y/N)	_____
Cache byte returned	_____

6. Homework Problem 9.12 (pg 850)
   Repeat Problem 9.11 for the following address:

Virtual address: `0x03a9`

A. Virtual address format:

13	12	11	10	9	8	7	6	5	4	3	2	1	0

B. Address translation:

Parameter	Value
VPN	_____
TLB index	_____
TLB tag	_____
TLB hit? (Y/N)	_____
Page fault? (Y/N)	_____
PPN	_____

C. Physical address format:

11	10	9	8	7	6	5	4	3	2	1	0

D. Physical memory reference:

Parameter	Value
Byte offset	_____
Cache index	_____
Cache tag	_____
Cache hit? (Y/N)	_____
Cache byte returned	_____

7. Homework Problem 9.13 (pg 851)

Repeat Problem 9.11 for the following address:

Virtual address: `0x0040`

A. Virtual address format:

13	12	11	10	9	8	7	6	5	4	3	2	1	0

B. Address translation:

Parameter	Value
VPN	_____
TLB index	_____
TLB tag	_____
TLB hit? (Y/N)	_____
Page fault? (Y/N)	_____
PPN	_____

C. Physical address format:

11	10	9	8	7	6	5	4	3	2	1	0

D. Physical memory reference:

Parameter	Value
Byte offset	_____
Cache index	_____
Cache tag	_____
Cache hit? (Y/N)	_____
Cache byte returned	_____

Created Wed 22 May 2013 2:52 PM PDT

Last Modified Wed 16 Jul 2014 9:58 AM PDT