CSEE W3827

Fundamentals of Computer Systems

Homework Assignment 6

Prof. Martha A. Kim
Columbia University
Due Apr 21, 2016 by 5:00 PM

- Write your name **and UNI** on each page of your solutions.
- Show your work.
- Note your collaborators.
- Turn in to the CSEE 3827 dropbox in the TA room.

1. (30 pts.) Complete the cycle-by-cycle execution trace of the min4 function when called on $a0 = 30, $a1 = 40, $a2 = 20, $a3 = 10.

```
min4:   add  $v0, $0, $a0
cmp_a1: sltu $t0, $a1, $v0
        beq  $0, $t0, cmp_a2
        add  $v0, $0, $a1
cmp_a2: sltu $t0, $a2, $v0
        beq  $0, $t0, cmp_a3
        add  $v0, $0, $a2
cmp_a3: sltu $t0, $a3, $v0
        beq  $0, $t0, done
        add  $v0, $0, $a2
done:   jr   $ra
```

   (a) (15 pts.) Assume a five-stage pipeline with early branch resolution but *no forwarding*. The only option to resolve hazards is to stall (i.e., insert bubbles).

add $v0, $0, $a0	F	D	X	M	W			
sltu $t0, $a1, $v0		F	D	D	D	X	M	W
...								

   (b) (15 pts.) Assume a five-stage pipeline with early branch resolution and *full forwarding* (i.e., W-X, M-X, M-D).

add $v0, $0, $a0	F	D	X	M	W	
sltu $t0, $a1, $v0		F	D	X	M	W
...						

2. (25 pts.) This question analyzes the following array sum function, whose first argument is a pointer to an array of integers and whose second argument is the length of the array.

```
add1:  add  $v0, $0, $0
beq1:  beq  $a1, $0, jr1
addi1: addi $a1, $a1, -1
lw1:   lw   $t0, 0($a0)
add2:  add  $v0, $v0, $t0
addi2: addi $a0, $a0, 4
beq2:  beq  $0, $0, beq1
jr1:   jr   $ra
```

(a) (5 pts.) Give execution counts for each instruction in the program.

(b) (5 pts.) Assuming a five stage pipeline with early branch resolution and full forwarding, indicate where the bubbles occur and how many there are (e.g., 1 between add1 and beq1).

(c) (5 pts.) What is the dynamic instruction count for this function when called on an array of length X?

(d) (5 pts.) What is the cycle count for this function when called on an array of length X?

(e) (5 pts.) What is the CPI of this function for very long arrays?

3. (20 pts.) Re-implement the array sum function to optimize performance. What is the new, optimized, CPI?

4. (25 pts.) Assume that the original array sum function is called on an array of length three starting at address A.

(a) (5 pts.) List the data address references the function call makes.

(b) (5 pts.) List the instruction address (i.e., PC) references the function call makes.

(c) (5 pts.) What is the hit rate of this call on an L1 data cache that is 256B, direct mapped, with 16B lines? Assume that A = 0x0000EFA0.

(d) (5 pts.) What is the hit rate of this call on an L1 instruction cache that is 64B, direct mapped, with 16B lines? Assume that the PC of the first instruction is 0x00FF000C.

(e) (5 pts.) What is the hit rate of this call on a unified L2 instruction and data cache that is 2-way set associative, 1024B capacity, with 16B lines?