Assignment 5 CS 3339 – Fall 2017 Name Daniel Ortiz

Due: Tues 11/14/17 @ 11:55pm netID: deo15

40 points (no late grace period) (email not long Axxxxx number)

All submissions must be written in very neat handwriting and scanned (or typed) and submitted in PDF format to TRACS with the filename of A5_netID.pdf. You may submit as many times as you like prior to the deadline; only the most recent submittal will be graded. All assignments must be submitted individually and reflect your own work; however, I encourage you to work in groups and discuss the problems with your classmates.

---

1) [4 points] What is the best CPI a superscalar CPU with n pipelines can achieve? What performance metric do we usually use instead of CPI when we discuss superscalar pipelines?

1/n
IPC

2) [4 points] Even without prefetching it is possible for the first-ever access to a word to hit in the cache. What allows this to happen?

Yes because of spatial locality and write allocate. Because there are usually multiple words in a block, the next read could be a hit.

3) [4 points] If higher associativity increases the hit ratio, why does it not always improve performance?

Requires all entries to be searched at once.
Comparator per entry (expensive).

4) [4 points] Given a direct mapped cache with one word per block and four blocks with the following accesses, circle the cycles/addresses that are cache hits. What is the hit ratio?

0x0,	0x18,	0xC,	0x0,	0x8,	0xC,	0x18,	0x20,	0x0,	0x8
0000,	0001_1000,	1100,	0000,	1000,	1100,	0001_1000,	0010_0000,	0000,	1000

$\frac{2}{10}$ Hit rate

0x0 [0000_0000] miss → block 0

0x18 [0001_1000] miss → block 1

0xC [0000_1100] miss → block 2

0x0 [0000_0000] **HIT** → block 0

0x8 [0000_1000] miss → block 1 OVERRIDE

0xC [0000_1100] **HIT** → block 2

0x18 [0001_1000] miss → block 1 OVERRIDE

0x20 [0010_0000] miss → block 0 OVERRIDE

0x0 [0000_0000] miss → block 0 OVERRIDE

0x8 [0000_1000] miss → block 1 OVERRIDE

5) [4 points] Assume a processor with a 2GHz clock. The cache has a base cache access time (including hit detection) of 1 clock cycle, and L1 miss penalty of 8 cycles, and an L2 miss penalty of 50 cycles. Assume that 6% of read accesses to the L1 data cache miss and that 25% of read accesses to the L2 miss. What is the average memory access time per load instruction? (State your answer in nanoseconds)

AMAT = 1 + (0.06)8 + (0.06)(0.25)50 = 2.23Cycles(0.5ns) **= 1.115ns**

6) [6 points] In the following C++ code segment for matrix-matrix multiply, specify for each of the matrices A, B, and C whether the matrix's accesses primarily have spatial, temporal, or no locality. (Assume accesses few than N words apart are considered to have spatial locality).

```
for(k = 0; k < N; k++){
  for(i = 0; i < N; i++){
    for(j = 0; j < N; j++){
      C[i][j] += A[i][k]* B[k][j];
    }
  }
}
```

A: Temporal

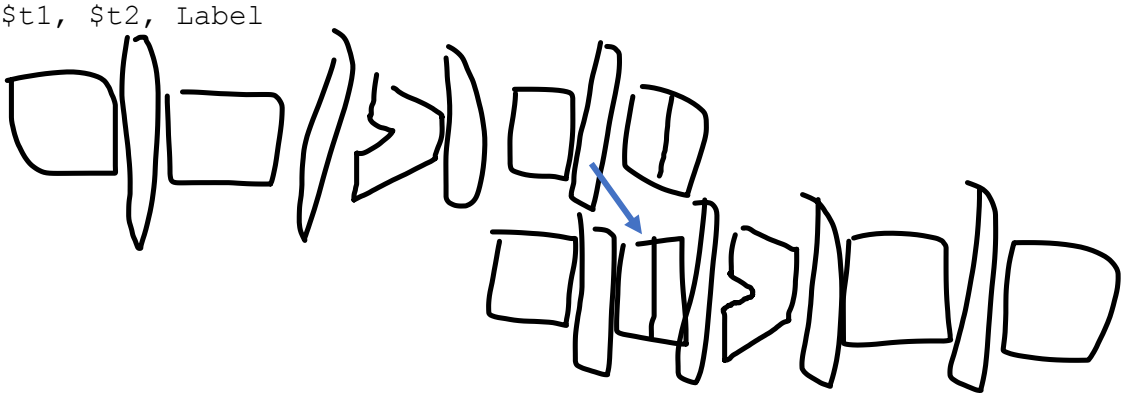
B: Spatial

C: Spatial

7) [4 points] Given a 5-stage MIPS pipeline implementation indicate which forwarding path will be used by the following sequence of instructions. Use the "??/??->?? notation where ??/?? designates the source flip/flop (e.g. EXE/MEM) and the last ?? indicates the cycle in which the forwarding path will be active.

```
sub $t2, $t4, $t1
and $t3, $t1, $t0
lw  $t5, 4($t1)
beq $t1, $t2, Label
```

MEM/WB -> ID



8) [10 points] Unroll the following loop four times and optimize the resulting instruction sequence. Assume the loop count is a multiple of four. Further assume the 5-stage MIPS pipeline with ID-stage branch resolution, and make sure that no bubbles need to be generated by the CPU. [Hint: There is a source dependency on $t1; be careful about bubbles from this!]

```
label: sw   $zero, 0,($t1)
       addi $t1, $t1, 4
       bne  $t1, $t2, label
```

```
sw          sw $zero, 0,($t1)
addi        sw $zero, 4,($t1)
sw          sw $zero, 8,($t1)
addi        addi $t1, $t1, 16
sw          sw $zero, -4,($t1)
addi        bne $t1, $t2, label
sw
addi
bne
```