Worksheet – 10 – CS2400 Chris Welch

1. In a Direct Map cache for the Block addresses: 00011_2 and 10101_2. Cache index in a 8 oneword block size cache = ___3___ and ___5___ (8 Line Cache)
   3 % 8 = 3 & 21 % 8 = 5
2. Given initial state of cache as the table on right, write if each address gets a Hit or Miss.
   Index = Index
   Tag = Tag

Index = Index
Tag = Tag

Address	Hit or Miss
10110	M
11010	M
10110	H
11010	H
10000	M
00011	M
10000	H
10010	M
10000	H

Index	V	Tag	Data
000	~~N~~ Y	10	10000
001	N		
010	~~N~~ Y	~~11~~ 10	10010
011	~~N~~ Y	00	00011
100	N		
101	N		
110	~~N~~ Y	10	10110
111	N		

No values Stored^

Stored values are a miss, if not contained.
if (Index == this.Index && Tag == this.Tag) HIT++;
else{ MISS++;}

3. If the miss rate of an instruction cache is 2% and the miss rate of the data cache is 4%. If a processor has a CPI of 2 without any memory stalls, and the miss penalty is 100 cycles for all misses, determine how much faster a processor would run with a perfect cache that never missed. Assume the frequency of all loads and stores is 36%.

I Miss Rate = 2%
Data Miss Rate = 4%
CPI = 2
Miss Penalty = 100 cycles
LDR/STR = 36% Frequency

Instruction Miss Cycles =
I × 2% × 100 = 2.00 I

Data Miss Cycles =
I × 36% × 4% × 100 = 1.44 I

Memory Stall Total =
2.00 I + 1.44 I = 3.44 I

CPI w/ Stalls
2 + 3.44 = 5.44

Perfect Cache = 2

$$\text{Performance} = \frac{\text{CPI stall}}{\text{CPI perfect}} = \frac{5.44}{2} = 2.72$$

Chris Welch

4. Find the AMAT for a processor with a 1 ns clock cycle time, a miss penalty of 20 clock cycles, a miss rate of 0.05 misses per instruction, and a cache access time (including hit detection) of 1 clock cycle. Assume that the read and write miss penalties are the same and ignore other write stalls.

AMAT = Time for hit + Miss Rate x penalty

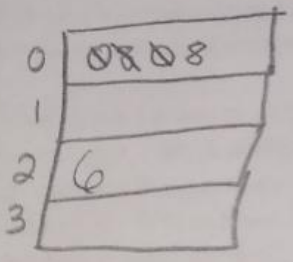
$1\text{ns} + 20 \times 0.05$

$= 2\text{ns}$

5. Assume there are three small caches, each consisting of four one-word blocks. One cache is direct-mapped, a second is two-way set-associative, and the third is fully associative. Find the number of misses for each cache organization given the following sequence of block addresses: **0, 8, 0, 6, and 8.**

n%e 4 = #Address

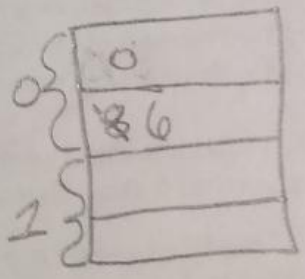
Direct Mapped



0%4 = 0 | Miss
8%4 = 0 | Miss
0%4 = 0 | Miss
6%4 = 2 | Miss
8%4 = 0 | Miss

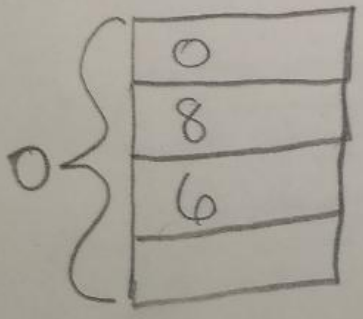
2-way-set-associative

n%e2= Address



0%2 = 0 | Miss
8%2 = 0 | Miss
0%2 = 0 | Hit
6%2 = 0 | Miss
8%2 = 0 | Miss

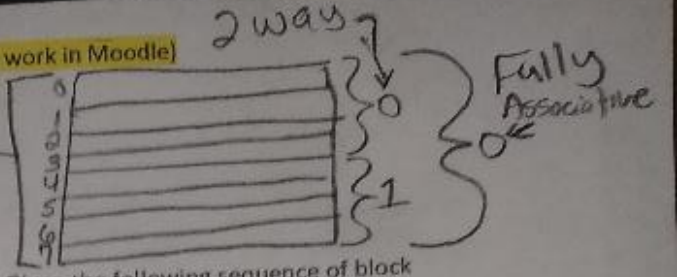
Fully Associative



0 = 0 | Miss
8 = 0 | Miss
0 = 0 | Hit
6 = 0 | Miss
8 = 0 | Hit

Chris Welch

6. Assume a cache with 8 one-word blocks. (Submit the work in Moodle)



Cache configuration:

1. Direct Mapped
2. Cache configuration: Two-way set-associative
3. Cache configuration: Fully associative

9.1. Assume a direct-mapped cache with 8-one word blocks. Given the following sequence of block addresses, indicate if each request results in a cache hit or miss: **1, 9, 6, 5, 1, 6.**

Cache block mapping:

- Block address 1 maps to cache block 1 • Block address 5 maps to cache block 5 • Block address 6 maps to cache block 6
- Block address 9 maps to cache block ____

Memory block 1 – (Hit or Miss) ? M
Memory block 9 – (Hit or Miss) ? M
Memory block 6 – (Hit or Miss) ? M
Memory block 5 – (Hit or Miss) ? M
Memory block 1 – (Hit or Miss) ? M
Memory block 6 – (Hit or Miss) ? H

1 % 8 = 1
9 % 8 = 1
6 % 8 = 6
5 % 8 = 5
1 % 8 = 1
6 % 8 = 6

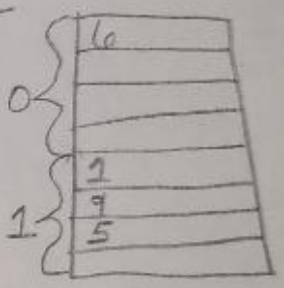
9.2. Assume a two-way set-associative cache with 8-one word blocks. Given the following sequence of block addresses, indicate if each request results in a cache hit or miss: **1, 9, 6, 5, 1, 6.** Cache set mapping:

- Block address 1 maps to cache set 1 • Block address 5 maps to cache set 1 • Block address 6 maps to cache set 0
- Block address 9 maps to cache set 1

Memory block 1 – (Hit or Miss) ? M
Memory block 9 – (Hit or Miss) ? M
Memory block 6 – (Hit or Miss) ? M
Memory block 5 – (Hit or Miss) ? M
Memory block 1 – (Hit or Miss) ? H
Memory block 6 – (Hit or Miss) ? H

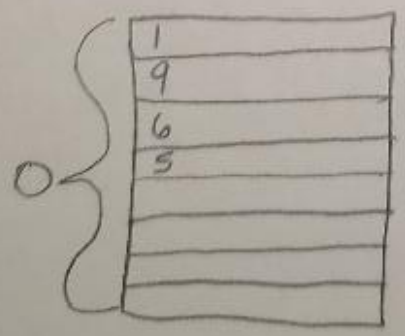
2 x 2

1 = 1
9 = 1
6 = 0
5 = 1
1 = 1
6 = 0



9.3 Assume a fully associative cache with 8-one-word blocks. Given the following sequence of block addresses, indicate if each request results in a cache hit or miss: **1, 9, 6, 5, 1, 6.**

Memory block 1 – (Hit or Miss) ? M
Memory block 9 – (Hit or Miss) ? M
Memory block 6 – (Hit or Miss) ? M
Memory block 5 – (Hit or Miss) ? M
Memory block 1 – (Hit or Miss) ? H
Memory block 6 – (Hit or Miss) ? H



n	n = 0
1	0
9	0
6	0
5	0
1	0
6	0