Computer Sciences Department
University of Wisconsin-Madison
CS/ECE 552 – Introduction to Computer Architecture
In-Class Exercise (03/10) SOLUTION

Answers to all questions should be uploaded on Canvas.

1. [1 point] (Twist on Check Yourself 5.9) Which of the following statements (if any) are generally true?

1. There is no way to reduce compulsory misses.
2. Fully associative caches have no conflict misses.
3. In reducing misses, associativity is more important than capacity.

Given that 2 is right, why is 3 wrong?

Solution:

2 is right because, by definition, fully associative caches remove all conflict misses. 3 is wrong because increasing cache capacity potentially helps all 3 C's (compulsory, capacity, conflict) by allowing more data to be stored in the cache at the same time – associativity only helps when we have conflicts.

2. [2 points] Assume a 4-way set-associative cache uses a 16-bit set index and has 12-bit tags for 32-bit memory addresses.

(a) [1 point] What is the block size?

Solution:

$2^{(32 - 16 - 12)}$ = 16-byte blocks

(b) [1 point] What is the cache capacity?

Solution:

16-byte blocks × 64K sets × 4 ways = 4MB capacity

3. [3 points] Show how a 1MB 16-way set-associative cache with 128-byte blocks will be indexed given a 32-bit memory address. In the table below, specify which bits will be used for the tag (**T**), set (**S**) and offset (**O**).

Solution:

Bit Index																															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	S	S	S	S	S	S	S	S	S	O	O	O	O	O	O	O

Since the blocks are 128B, there are $\log_2(128) = 7$ bits for the offset – these are the lowest 7 bits.

There are 1MB/128B = 8KB entries in our cache (each entry is 128B). Given that we have 16 sets, this means there are 8K/16 = 512 sets, each with 16 entries in them. To index into these sets, we need $\log_2(512) = 9$ bits. These are the next 9 bits.

The remaining bits are the tag bits.

4. [4 points] Consider the following sequence of accesses to memory blocks in one set of a 4-way set-associative cache.

(a) [3 points] Fill in the table below assuming LRU replacement. When there are unpopulated ways, assume they get filled in from left to right.

Solution:

Block	Way Contents Before	Way Contents After	LRU Before	LRU After
A	[-, -, -, -]	[A, -, -, -]	--	--
B	[A, -, -, -]	[A, B, -, -]	--	--
C	[A, B, -, -]	[A, B, C, -]	--	--
D	[A, B, C, -]	[A, B, C, D]	--	A
C	[A, B, C, D]	[A, B, C, D]	A	A
E	[A, B, C, D]	[E, B, C, D]	A	B
A	[E, B, C, D]	[E, A, C, D]	B	D
C	[E, A, C, D]	[E, A, C, D]	D	D
D	[E, A, C, D]	[E, A, C, D]	D	E
B	[E, A, C, D]	[B, A, C, D]	E	A
E	[B, A, C, D]	[B, E, C, D]	A	C
A	[B, E, C, D]	[B, E, A, D]	C	D
C	[B, E, A, D]	[B, E, A, C]	D	B

(b) [1 point] What is the miss rate?

Solution:

10 misses out of 13 accesses → 77%