Lecture: Cache Hierarchies

- Topics: cache access basics/examples
The Cache Hierarchy
Problem 1

• Memory access time: Assume a program that has cache access times of 1-cyc (L1), 10-cyc (L2), 30-cyc (L3), and 300-cyc (memory), and MPKIs of 20 (L1), 10 (L2), and 5 (L3). Should you get rid of the L3?
Problem 1

• Memory access time: Assume a program that has cache access times of 1-cyc (L1), 10-cyc (L2), 30-cyc (L3), and 300-cyc (memory), and MPKIs of 20 (L1), 10 (L2), and 5 (L3). Should you get rid of the L3?

With L3: $1000 + 10 \times 20 + 30 \times 10 + 300 \times 5 = 3000$
Without L3: $1000 + 10 \times 20 + 10 \times 300 = 4200$
Accessing the Cache

- Byte address
- 101000
- Offset

8-byte words

Data array

Sets

Direct-mapped cache: each address maps to a unique address

8 words: 3 index bits
The Tag Array

Direct-mapped cache: each address maps to a unique address

8-byte words

Tag array

Tag

Compare

Data array

Byte address

101000

Tag

8-byte words
Increasing Line Size

A large cache line size → smaller tag array, fewer misses because of spatial locality

32-byte cache line size or block size
Associativity

Set associativity → fewer conflicts; wasted power because multiple data and tags are read.

Byte address

10100000

Tag array

Tag

Compare

Way-1

Way-2

Data array
Problem 2

• Assume a direct-mapped cache with just 4 sets. Assume that block A maps to set 0, B to 1, C to 2, D to 3, E to 0, and so on. For the following access pattern, estimate the hits and misses:

A B B E C C A D B F A E G C G A
Problem 2

- Assume a direct-mapped cache with just 4 sets. Assume that block A maps to set 0, B to 1, C to 2, D to 3, E to 0, and so on. For the following access pattern, estimate the hits and misses:

A B B E C C A D B F A E G C G A
M MH MM H MM HM HMM M M M M
Problem 3

• Assume a 2-way set-associative cache with just 2 sets. Assume that block A maps to set 0, B to 1, C to 0, D to 1, E to 0, and so on. For the following access pattern, estimate the hits and misses:

A B B E C C A D B F A E G C G A
Problem 3

• Assume a 2-way set-associative cache with just 2 sets. Assume that block A maps to set 0, B to 1, C to 0, D to 1, E to 0, and so on. For the following access pattern, estimate the hits and misses:

A B B E C C A D B F A E G C G A
M MH M MH MM HM HMM M H M
Problem 4

• 64 KB 16-way set-associative data cache array with 64 byte line sizes, assume a 40-bit address

• How many sets?

• How many index bits, offset bits, tag bits?

• How large is the tag array?

Equations:
Data array size (cache size) = \#sets \times \#ways \times \text{blocksize}
Tag array size = \#sets \times \#ways \times \text{tagsize}
Index bits = \log_2 (\#sets)
Offset bits = \log_2 (\text{blocksize})
Tag bits + index bits + offset bits = address width
Problem 4

- 64 KB 16-way set-associative data cache array with 64 byte line sizes, assume a 40-bit address

- How many sets? 64

- How many index bits (6), offset bits (6), tag bits (28)?

- How large is the tag array (28 Kb)?
Problem 5

• 8 KB fully-associative data cache array with 64 byte line sizes, assume a 40-bit address

• How many sets? How many ways?

• How many index bits, offset bits, tag bits?

• How large is the tag array?
Problem 5

• 8 KB fully-associative data cache array with 64 byte line sizes, assume a 40-bit address

• How many sets (1) ? How many ways (128) ?

• How many index bits (0), offset bits (6), tag bits (34) ?

• How large is the tag array (544 bytes) ?
Types of Cache Misses

• Compulsory misses: happens the first time a memory word is accessed – the misses for an infinite cache

• Capacity misses: happens because the program touched many other words before re-touching the same word – the misses for a fully-associative cache

• Conflict misses: happens because two words map to the same location in the cache – the misses generated while moving from a fully-associative to a direct-mapped cache

• Sidenote: can a fully-associative cache have more misses than a direct-mapped cache of the same size?