Lecture 22: Cache Hierarchies, Memory

• Today’s topics:
  - Cache hierarchies
  - DRAM main memory
  - Virtual memory
Locality

• Why do caches work?
  ▪ Temporal locality: if you used some data recently, you will likely use it again
  ▪ Spatial locality: if you used some data recently, you will likely access its neighbors

• No hierarchy: average access time for data = 300 cycles

• 32KB 1-cycle L1 cache that has a hit rate of 95%:
  average access time = 0.95 \times 1 + 0.05 \times (301) = 16 \text{ cycles}
Accessing the Cache

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

8-byte words

Sets

Data array

Offset

Byte address

101000

8 words: 3 index bits
The Tag Array

Direct-mapped cache: each address maps to a unique address.
Example Access Pattern

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

Assume that addresses are 8 bits long
How many of the following address requests are hits/misses?
4, 7, 10, 13, 16, 68, 73, 78, 83, 88, 4, 7, 10…

Tag array

Compare

8-byte words

Data array
Increasing Line Size

A large cache line size $\rightarrow$ smaller tag array, fewer misses because of spatial locality

Tag array

Offset

Data array

32-byte cache line size or block size

Byte address

10100000

Tag
Associativity

- Byte address
- Tag array
- Compare
- Tag
- Data array
- Way-1
- Way-2

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

How many offset/index/tag bits if the cache has 64 sets, each set has 64 bytes, 4 ways

Tag array Compare Data array

10100000

Tag

Byte address

Way-1 Way-2
Example

• 32 KB 4-way set-associative data cache array with 32 byte line sizes

• How many sets?

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

• How large is the tag array?
Title

• Bullet