CS61C
Fall 2026

Caches
Discussion 10

1 Cache T/I/O

1.1 Assume we have a 32-bit address space with a 32B, direct-mapped cache with 8B blocks. Of the 32 bits in each address, which bits do we use to find the tag, index, and offset of the cache?

1.2 Assume that we have the same cache scheme as the previous part (a 32-bit address space with a 32B, direct-mapped cache with 8B blocks). Do the following:

- Decode the tag, index, and offset bits for each address
- Classify each of the following accesses as a cache hit (H) or a cache miss (M).
  - For any misses, list out what type of miss it is (Compulsory (C), Noncompulsory (NC)).

Tip: Use the space below to draw out your cache!

Address	T/I/O	Hit / Miss (C / NC)
`0x00000004`		
`0x00000005`		
`0x00000068`		
`0x000000C8`		
`0x00000068`		
`0x000000DD`		
`0x00000045`		
`0x000000CF`		
`0x000000F3`		

2 Cache Associativity

2.1 Here's some practice involving a **2-way set associative** cache. Assume we have an 8-bit address space with a 32B, 2-way set associative cache with 8B blocks.

- Decode the tag, index, and offset bits for each address
- Classify each of the following accesses as a cache hit (H) or a cache miss (M).
  - For any misses, list out what type of miss it is (Compulsory (C), Noncompulsory (NC)).

Assume that we have an LRU replacement policy (in general, this is not always the case).

Address	T/I/O	Hit / Miss (C / NC)
`0b0000 0100`		
`0b0000 0101`		
`0b0110 1000`		
`0b1100 1000`		
`0b0110 1000`		
`0b1101 1101`		
`0b0100 0101`		
`0b0000 0100`		
`0b0011 0000`		
`0b1100 1011`		
`0b0100 0010`		

2.2 What is the hit rate of our above accesses?

3 Code Analysis

Given the follow chunk of code, analyze the hit rate given that we have a byte-addressed computer with a total memory of **1 MiB**. It also features a **16 KiB** Direct-Mapped cache with **1 KiB** blocks. Assume that your cache begins cold.

```
#define NUM_INTS 8192   // 2^13
int A[NUM_INTS];        // A lives at 0x10000
int i, total = 0;
for (i = 0; i < NUM_INTS; i += 128) {
    A[i] = i;           // Line 1
}
for (i = 0; i < NUM_INTS; i += 128) {
    total += A[i];      // Line 2
}
```

3.1 How many bits make up a memory address on this computer?

3.2 What is the T/I/O breakdown?

3.3 Calculate the cache hit rate for the line marked Line 1:

3.4 Calculate the cache hit rate for the line marked Line 2 (Note that the cache still retains contents from earlier lines):

4 Cache Performance

Recall that AMAT stands for Average Memory Access Time. The main formula for it is:

$$\text{AMAT} = \text{Hit Time} + \text{Miss Rate} * \text{Miss Penalty}$$

4.1 Assume clock speed is 1GHz. Suppose your system takes 100 cycles to access main memory. We decide to add a cache with a measured hit time of 25 cycles and miss rate of 25%. What is the average memory access time of the system in nanoseconds?

4.2 In a new 2-level cache system, after 100 total accesses to the cache system, we find that the L2$ (L2 Cache) ended up missing 20 times. What is the global miss rate of L2$?

4.3 Given the system from the previous subpart (100 total accesses, 20 L2$ misses), if L1$ had a local miss rate of 50%, what is the local miss rate of L2$?

For the following subparts, suppose we have a new system that consists of:

1. An L1$ that has a hit time of 2 cycles and a local miss rate of 20%
2. An L2$ that has a hit time of 15 cycles and has a global miss rate of 5%
3. Main memory where accesses take 100 cycles

4.4 What is the local miss rate of L2$?

4.5 What is the AMAT of the system in cycles?

4.6 Suppose we want to reduce the AMAT of the system to 8 cycles or lower by adding in a L3$. If the L3$ has a local miss rate of 30%, what is the largest hit time that L3$ can have?