Direct-Mapped Cache Lookup

Ex:

Cache Size = 16 KB
Block Size = 32 bytes
# blocks = \( \frac{16 \text{ KB}}{32} = 512 \)

32-bit address
18
9
5

hit?
(2-Way) Set Associative Cache Lookup

Ex: Cache Size = 16 KB
Block Size = 32 bytes
# Sets = \( \frac{16 \times 1024}{32 \times 2} = 256 \)
Fully Associative Cache Lookup

Ex: Cache Size = 16 KB

Block Size = 32 bytes

# blocks = \( \frac{16K}{32} = 512 \)
Victim Cache

Motivation:
Conflict misses usually isolated to a few sets

<table>
<thead>
<tr>
<th>Set</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

num of blocks that map to each set.

On access, check both caches (in parallel)
- Eviction from primary cache → victim cache
- Victim cache hit → primary cache
- If both caches miss, fetch from main memory → primary cache.
Multi-Level Caches

- Goal: Fast + large cache
- As you go down hierarchy:
  - Larger cache
  - Larger block size
  - Higher associativity

- Inclusion

Avg Mem. Acc. Time = \( \text{hit}_{L_1} + \text{miss rate}_{L_1} (\text{hit}_{L_2} + \text{miss rate}_{L_2} (\text{hit}_{L_3} + \text{miss rate}_{L_3} \text{miss penalty})) \)