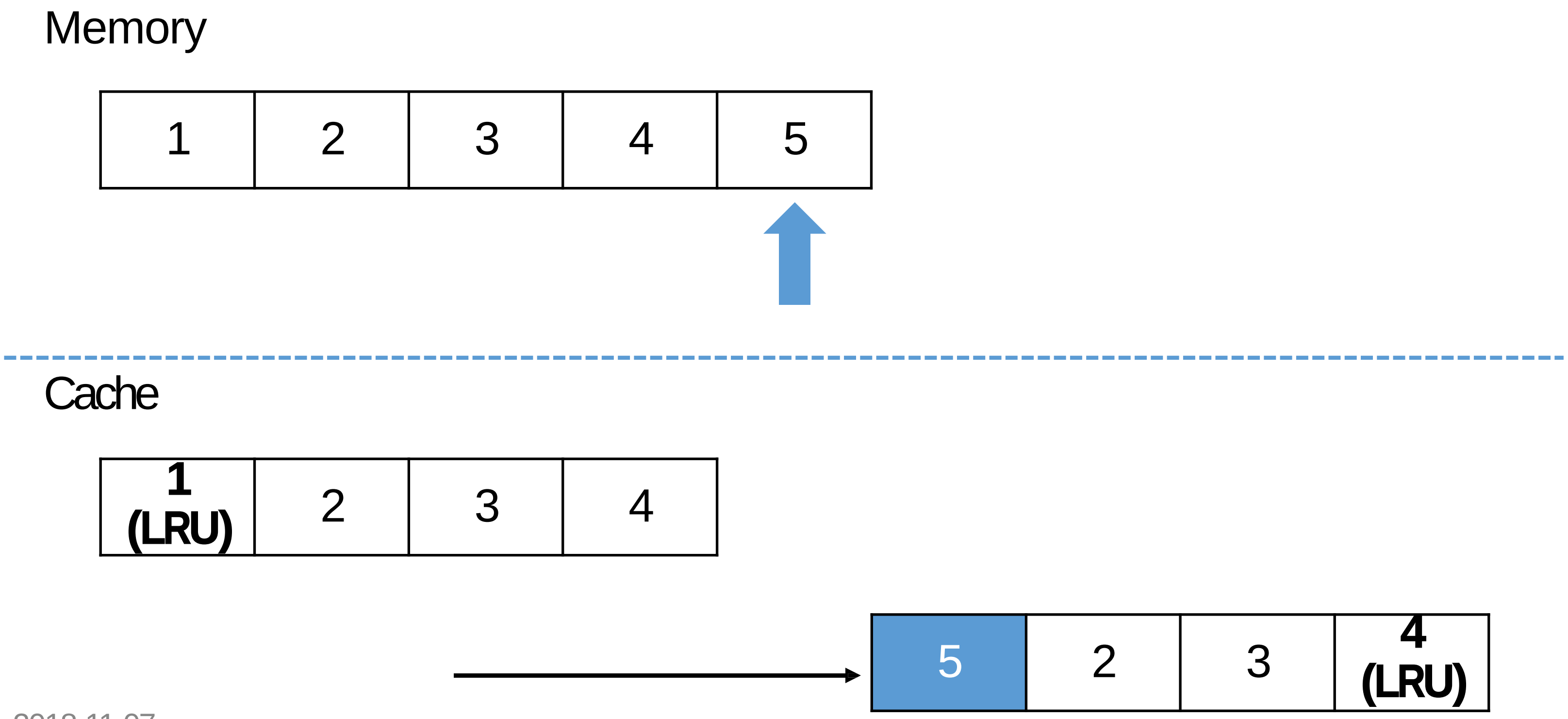
Lab 08
Cache (Part B)
- Optimization -

2018-11-07

Cache Replacement Policy

- **Least Recently Used (LRU)**
  - Replace the cache block which was used least recently



Memory

1	2	3	4	5

Cache

1 (LRU)	2	3	4

5	2	3	4 (LRU)

Types of Cache Misses

- **Cold (compulsory) miss**
  - The first access to a block has to be a miss as the corr esponding block would not have been cached yet.
- **Conflict miss**
  - Conflict misses occur when the level *k* cache is large enough, but multiple data objects all map to the same level *k* block
- **Capacity miss**
  - Occurs when the set of active cache blocks (working set) is larger than the cache

Hit Ratio

- The percentage of accesses that result in cache hits
- Example
  - 32 bytes direct mapped cache with a block size of 16 bytes

Row-major order

int A[4][4]

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

Cache

9	10	11	12
13	14	15	16

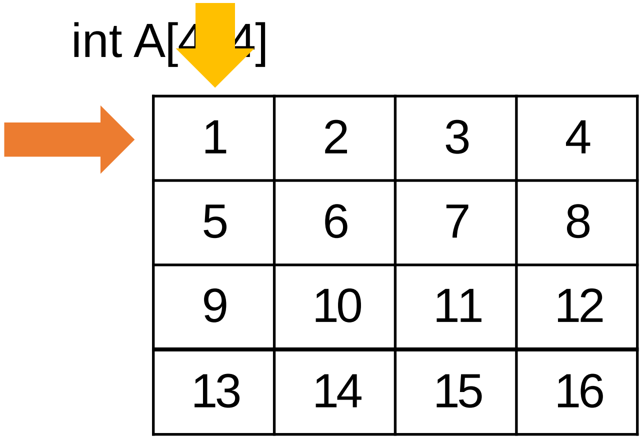
Hit Ratio: 3/4

Hit Ratio

- The percentage of accesses that result in cache hits
- Example
  - 32 bytes direct mapped cache with a block size of 16 bytes

Column-major order

int A[4][4]



1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

Cache

1	2	3	4

Cachelab Part B

- Optimize matrix transpose (A → $A^{\mathbf{T}}$)
  - Write the efficient code with the highest hit ratio (i.e. minimize the cache miss)

- Reference README file
  - Notice ‘**Blocking**’ technique
  - You would be better to think about diagonal entries.