CSE3666

Homework 8 Solutions

1

a)

For each block, tag and index are known. The bits in the block offset are all 1's or all 0's.

b)

Find cache index.
Compare the tags. Note that all cache blocks are valid in this question.
If it is a miss, update the tag (and other information for the cache block). The real cache will load data from memory.

2

See the examples in slides. The numbers are checked in Gradescope.

In 2g, the percentage is calculated by dividing the number of bits for tags and status bits by the total number of bits. The number of blocks can be canceled out before division.

3

See the examples in slides. The numbers are checked in Gradescope.

4

a)

$1 + 0.1 * 80 = 9$

b)

$1 + 0.05 * 80 = 5$

c)

$0.35 * 0.1 * 80 = 2.8$

d)

$1 * 0.05 * 80 = 4$

e)

$1.6 + 2.8 + 4 = 8.4$

f)

$8.4 / 1.6 = 5.25$

5

Since each block can hold 16 elements. D[0] and D[15] are loaded into cache.
Note that the lower 8 bits of D[0]'s are 0. So D[0] starts a new block.
Therefore, when D[1] is accessed, it is a hit.

Accessing D[1] through D[15] is cache hit. The first miss will be D[16].

D[0] is evicted when CPU accesses a word when the cache is filled. The cache size is 64KiB. Each element in the array is 2 bytes. It takes 64 * 1024 / 2 = 32768 elements to fill up the cache. The last block is loaded into cache when D[32752] is accessed.

D[0] is evicted when D[32768] is accessed.

On every cache miss, a block of 32 bytes, which consists of 16 elements in the array, is loaded into cache. The 15 references after a miss will be hits. So the miss rate is 1/16 = 6.25%.

If the block size is changed to 64 bytes, there is a miss every 32 accesses. The miss rate is 1/32.

Note that the analysis is similar to what we do in lab 9. However, the cache is not large enough to store the entire array here. Also, each element in the array is accessed only once.