CST 334 (Operating Systems)

Dr. Glenn Bruns

# Lab: Replacement Policies

1. Does cache replacement occur following a cache hit, or a cache miss?
2. Let's look at how LRU works on a looping workload, where the cache is smaller than the loop size. Fill in the blanks, using the LRU policy and cache size of 3. What is the cache hit ratio?

|  |  |  |  |
| --- | --- | --- | --- |
| Access | Hit/Miss | Evict | Resulting Cache State |
| 1 | miss | - | 1 |
| 2 | miss | - | 1,2 |
| 3 | miss | - | 1,2,3 |
| 4 | miss | yes | 2,3,4 |
| 1 | m | 2 | 3,4,1 |
| 2 | m | 3 | 4,1,2 |
| 3 | m | 4 | 1,2,3 |
| 4 | m | 1 | 2,3,4 |
| 1 | M | 2 | 3,4,1 |
| 2 | M | 3 | 4,1,2 |
| 3 | M | 4 | 1,2,3 |

1. Let's compare FIFO, LRU, and Belady's optimal policy on a single example workload. FIll in the blanks, assuming cache size of 3 and **FIFO policy**. After you're done, calculate the cache hit rate (a number between 0 and 1).

|  |  |  |  |
| --- | --- | --- | --- |
| Access | Hit/Miss | Evict | Resulting Cache State |
| 1 | miss | - | 1 |
| 3 | miss | - | 1,3 |
| 0 | miss | - | 1,3,0 |
| 1 | hit | - | 1,3,0 |
| 1 | hit | - | 1,3,0 |
| 1 | hit | - | 1,3,0 |
| 4 | Miss | 1 | 3,0,4 |
| 2 | miss | 3 | 0,4,2 |
| 0 | hit | - | 0,4,2 |
| 0 | hit | - | 0,4,2 |
| 3 | miss | 0 | 4,2,3 |

1. Fill in the blanks, assuming cache size of 3 and **LRU (least recently used) policy**. After you're done, calculate the cache hit rate.

|  |  |  |  |
| --- | --- | --- | --- |
| Access | Hit/Miss | Evict | Resulting Cache State |
| 1 | miss | - | 1 |
| 3 | miss | - | 1,3 |
| 0 | miss | - | 1,3,0 |
| 1 | hit | - | 3,0,1 |
| 1 | hit | - | 3,0,1 |
| 4 | miss | 3 | 0,1,4 |
| 1 | Hit | - | 0,1,4 |
| 2 | Miss | 0 | 1,4,2 |
| 0 | miss | 1 | 4,2,0 |
| 0 | Hit | - | 4,2,0 |
| 3 | Miss | 4 | 2,0,3 |

1. Fill in the blanks, assuming cache size of 3 and Belady's optimal policy. After you're done, calculate the cache hit rate.

|  |  |  |  |
| --- | --- | --- | --- |
| Access | Hit/Miss | Evict | Resulting Cache State |
| 1 | miss | - | 1 |
| 3 | miss | - | 1,3 |
| 0 | miss | - | 1,3,0 |
| 1 | hit | - | 1,3,0 |
| 1 | hit | - | 1,3,0 |
| 4 | miss | 3 | 1,0,4 |
| 1 |  |  |  |
| 2 |  |  |  |
| 0 |  |  |  |
| 0 |  |  |  |
| 3 |  |  |  |

1. Have an encounter with the mysterious “Belady anomaly”. Work out this example using the FIFO policy and a cache size of 3. Compute the cache hit rate.

|  |  |  |  |
| --- | --- | --- | --- |
| Access | Hit/Miss | Evict | Resulting Cache State |
| 1 | miss | - | 1 |
| 2 | miss | - | 1,2 |
| 3 | miss | - | 1,2,3 |
| 4 | miss | 1 | 2,3,4 |
| 1 | miss | 2 | 3,4,1 |
| 2 | miss | 3 | 4,1,2 |
| 5 |  |  |  |
| 1 |  |  |  |
| 2 |  |  |  |
| 3 |  |  |  |
| 4 |  |  |  |
| 5 |  |  |  |

1. If we increase the cache size, what do you expect should happen to the hit rate, assuming the same accesses are made? Work out this example, which is the same as the last, except you should use a cache size of 4. Compute the cache hit rate.

|  |  |  |  |
| --- | --- | --- | --- |
| Access | Hit/Miss | Evict | Resulting Cache State |
| 1 | miss | - | 1 |
| 2 | miss | - | 1,2 |
| 3 | miss | - | 1,2,3 |
| 4 | miss | - | 1,2,3,4 |
| 1 | hit | - | 1,2,3,4 |
| 2 | hit | - | 1,2,3,4 |
| 5 |  |  |  |
| 1 |  |  |  |
| 2 |  |  |  |
| 3 |  |  |  |
| 4 |  |  |  |
| 5 |  |  |  |

What is the change in cache hit rate between this example and the previous one? How can the hit rate decrease when the cache gets bigger? Look at the example carefully and try to understand how the hit rate decreased.

1. If you still have time, work out all the examples above using the OSTEP simulator.

Hints:

1. If there was a cache hit, is there any good reason to change what's in the cache?
2. The cache hit rate is 0.
3. You can get the answer to the first problem by running the OSTEP simulator named HW-Paging-Policy using the following command line parameters:

./paging-policy.py -s 131 -p FIFO -m 5 -c -n 11 # uses FIFO replacement policy

The answer to problem 1:

|  |  |  |  |
| --- | --- | --- | --- |
| Access | Hit/Miss | Evict | Resulting Cache State |
| 1 | miss | - | 1 |
| 3 | miss | - | 1,3 |
| 0 | miss | - | 1,3,0 |
| 1 | hit | - | 1,3,0 |
| 1 | hit | - | 1,3,0 |
| 4 | miss | 1 | 3,0,4 |
| 1 | miss | 3 | 0,4,1 |
| 2 | miss | 0 | 4,1,2 |
| 0 | miss | 4 | 1,2,0 |
| 0 | hit | - | 1,2,0 |
| 3 | miss | 1 | 2,0,3 |

1. You can use the OSTEP simulator to get the answer:

./paging-policy.py -s 131 -p LRU -m 5 -c -n 11

The answer is:

|  |  |  |  |
| --- | --- | --- | --- |
| Access | Hit/Miss | Evict | Resulting Cache State |
| 1 | miss | - | 1 |
| 3 | miss | - | 1,3 |
| 0 | miss | - | 1,3,0 |
| 1 | hit | - | 3,0,1 |
| 1 | hit | - | 3,0,1 |
| 4 | miss | 3 | 0,1,4 |
| 1 | hit | - | 0,4,1 |
| 2 | miss | 0 | 4,1,2 |
| 0 | miss | 4 | 1,2,0 |
| 0 | hit | - | 1,2,0 |
| 3 | miss | 1 | 2,0,3 |

1. You can get the answer using the simulator:

./paging-policy.py -s 131 -p OPT -m 5 -c -n 11

The answer is:

|  |  |  |  |
| --- | --- | --- | --- |
| Access | Hit/Miss | Evict | Resulting Cache State |
| 1 | miss | - | 1 |
| 3 | miss | - | 1,3 |
| 0 | miss | - | 1,3,0 |
| 1 | hit | - | 1,3,0 |
| 1 | hit | - | 1,3,0 |
| 4 | miss | 3 | 1,0,4 |
| 1 | hit | - | 1,0,4 |
| 2 | miss | 4 | 1,0,2 |
| 0 | hit | - | 1,0,2 |
| 0 | hit | - | 1,0,2 |
| 3 | miss | 2 | 1,0,3 |

1. The cache hit rate is 0.25. The sequence of evicts is -,-,-,1,2,3,4,-,-,1,2,-
2. The cache hit rate is 0.167. The sequence of evicts is -,-,-,-,-,1,2,3,4,5,1
3. See hint 2 above.