Beyond Physical Memory: Policies

Annalise Tarhan

February 17, 2021

1 Generate random addresses with the following arguments. Change the policy from FIFO, to LRU, to OPT. Compute whether each access in said address traces are hits or misses.

1.1 -s 0 -n 10

FIFO:
Miss - 8
Miss - 8,7
Miss - 8,7,4
Miss - 7,4,2
Miss - 4,2,5
Hit - 4,2,5
Miss - 2,5,7
Miss - 5,7,3
Miss - 7,3,4
Miss - 3,4,5

LRU:
Miss - 8
Miss - 8,7
Miss - 8,7,4
Miss - 7,4,2
Miss - 4,2,5
Hit - 2,5,4
Miss - 5,4,7
Miss - 4,7,3
Hit - 7,3,4
Miss - 3,4,5

OPT:
Miss - 8

Miss - 8,7
Miss - 8,7,4
Miss - 2,7,4
Miss - 5,7,4
Hit - 5,4,7
Hit - 7,4,5
Miss - 3,4,5
Hit - 3,4,5
Hit - 3,4,5

1.2 -s 1 -n 10

FIFO:
Miss - 1
Miss - 1,8
Miss - 1,8,7
Miss - 8,7,2
Miss - 7,2,4
Hit - 7,2,4
Miss - 2,4,6
Miss - 4,6,7
Miss - 6,7,0
Hit - 6,7,0

LRU:
Miss - 1
Miss - 1,8
Miss - 1,8,7
Miss - 8,7,2
Miss - 7,2,4
Hit - 7,2,4
Miss - 2,4,6
Miss - 4,6,7
Miss - 6,7,0
Hit - 6,7,0

OPT:
Miss - 1
Miss - 1,8
Miss - 1,8,7
Miss - 8,2,7
Miss - 2,7,4
Hit - 2,4,7
Miss - 4,6,7
Hit - 4,6,7
Miss - 6,7,0

Hit - 6,7,0

1.3 -s 2 -n 10

FIFO:
Miss - 9
Hit - 9
Miss - 9,0
Hit - 9,0
Miss - 9,0,8
Miss - 0,8,7
Miss - 8,7,6
Miss - 7,6,3
Hit - 7,6,3
Hit - 7,6,3

LRU:
Miss - 9
Hit - 9
Miss - 9,0
Hit - 9,0
Miss - 9,0,8
Miss - 0,8,7
Miss - 8,7,6
Miss - 7,6,3
Hit - 7,3,6
Hit - 7,3,6

OPT:
Miss - 9
Hit - 9
Miss - 9,0
Hit - 9,0
Miss - 9,0,8
Miss - 0,8,7
Miss - 8,7,6
Miss - 7,3,6
Hit - 7,3,6
Hit - 7,3,6

2 For a cache of size 5, generate worst-case address reference streams for each of the following policies: FIFO, LRU, and MRU. For the worst-case reference streams, how much bigger of a cache is needed to improve performance dramatically and approach OPT?

FIFO:
1,2,3,4,5,6,1,2,3,4,5,6
LRU:
1,2,3,4,5,6,1,2,3,4,5,6
MRU:
1,2,3,4,5,6,5,6,5,6,5,6
For each example, a cache with one more space would give performance equal to OPT

3 Generate a random trace. How would you expect different policies to perform on such a trace?

{3,2,5,0,5,3,2,2,3,2,1}
With a cache size of 3, FIFO and LRU would both perform significantly worse than optimal, since it would discard the 2 and the 3 sooner than it should, holding on to 5 and 0 unnecessarily. MRU would also suffer, and random is unpredictable. LFU would do as well as LRU.

4 Now generate a trace with some locality. How can you generate such a trace? How does LRU perform on it? How much better than RAND is LRU? How does CLOCK do? How about CLOCK with different numbers of clock bits?

In this case, temporal locality is more relevant, so references to the same address should be close together. Given a completely sorted list, though, RAND does exactly as well as LRU. With a few non-sorted arguments inserted, RAND actually does better. CLOCK is similar.

5 Skipped