CIS 452 - Operating Systems Concepts

Nathan Bowman

Images taken from Silberschatz book

Optimal Replacement and LRU

FIFO replacement is simple, but does not account for usage and suffers from Belady's anomaly

Question becomes, how much can we improve over FIFO?

If FIFO gives 1/100,000 page fault rate, is it worth the effort to develop new algorithm?

What about 1/500,000?

Important to have some kind of benchmark to compare against

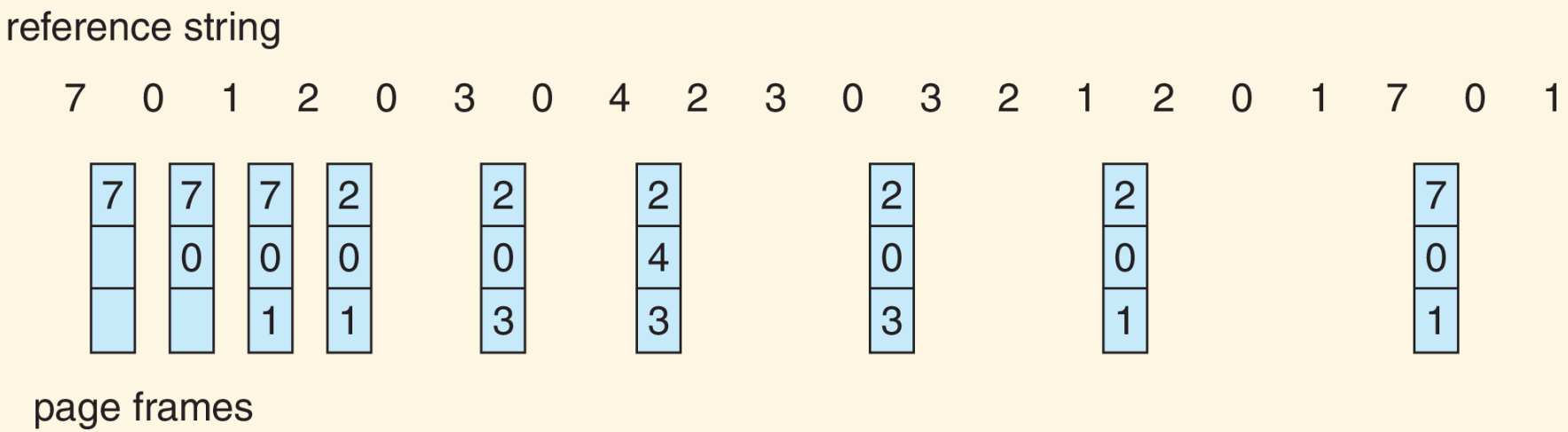
Ideally, compare to some kind of bound so it is known how close to optimal algorithm performs

Bound does not need to be attainable to be useful for comparison

This applies to anything you study, not just page replacement

Optimal page-replacement algorithm (OPT) is known and is quite simple

Replace page that will not be used for longest time


reference string
7 0 1 2 0 3 0 4 2 3 0 3 2 1 2 0 1 7 0 1
7 7 7 2 2 2 2 2 7
0 0 0 0 4 0 0 0
1 1 3 3 3 1 1
page frames

Properties of OPT:

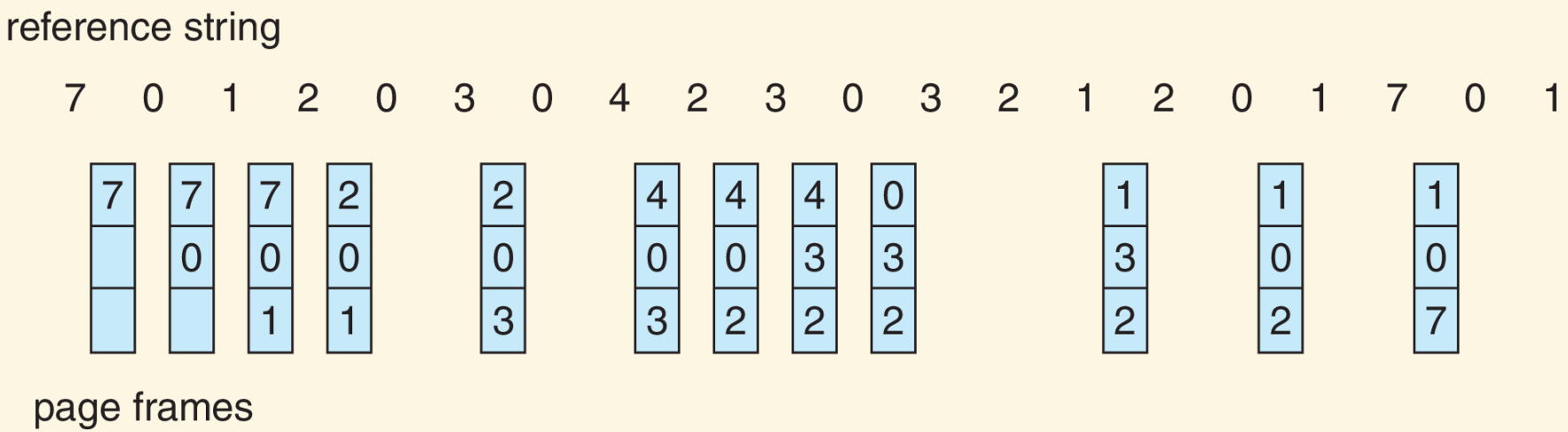
- provably minimal for fixed number of frames
- does not suffer from Belady's anomaly

However, requires knowledge of future, so is not able to be implemented in practice

Instead of looking forward to when page will be used next, look backward to when it was last used

Hopefully recent past is effective predictor of near future

Least-Recently Used (LRU) replacement chooses page that has not been read from or written to for longest time as victim page


reference string
7 0 1 2 0 3 0 4 2 3 0 3 2 1 2 0 1 7 0 1
page frames

LRU replacement, like OPT, cannot suffer from Belady's anomaly

Consider increasing number of frames from n to n + 1

In first scenario, n most-recently-used pages are kept in memory

In second scenario, n + 1 most-recently-used pages are kept in memory

n pages from first scenario are clearly subset of n + 1 pages from second scenario

Adding more frames therefore cannot hurt performance

Unlike FIFO, LRU considers whether page has been used, rather than just age

Makes for an effective algorithm -- variants of LRU are often used in actual systems

Tradeoff is that more information must be stored about each page

Details of implementing LRU will be considered in future lectures