

**9.4** Consider the following page-replacement algorithms. Rank these algorithms on a five-point scale from “bad” to “perfect” according to their page-fault rate. Separate those algorithms that suffer from Belady’s anomaly from those that do not.

a. LRU replacement

Rank: 3

Suffers from Belady’s: no

b. FIFO replacement

Rank: 1 (worst)

Suffers from Belady’s: yes

c. Optimal replacement

Rank: 4 (best but not feasible)

Suffers from Belady’s: no

d. Second-chance replacement

Rank: 2

Suffers from Belady’s: yes because if all reference bits are set then it behaves like FIFO.

NOTE: 5 (perfect) doesn’t exist.

**9.8** Consider the following page reference string:

1, 2, 3, 4, 2, 1, 5, 6, 2, 1, 2, 3, 7, 6, 3, 2, 1, 2, 3, 6.

How many page faults would occur for the following replacement algorithms, assuming one, two, three, four, five, six, and seven frames? Remember that all frames are initially empty, so your first unique pages will cost one fault each.

- LRU replacement
- FIFO replacement
- Optimal replacement

Frames	LRU	FIFO	Optimal
1	20	20	20
2	18	18	15
3	15	16	11

4	10	14	8
5	8	10	7
6	7	10	7
7	7	7	7

**9.28** Consider a demand-paging system with the following time-measured utilizations:

CPU utilization 20%

Paging disk 97.7%

Other I/O devices 5%

For each of the following, say whether it will (or is likely to) improve CPU utilization. Explain your answers.

- a. Install a faster CPU: No. The CPU utilization is already low. The problem here is Thrashing. A faster CPU will solve nothing.
- b. Install a bigger paging disk: No. Because the problem is thrashing, the problem is caused by not enough memory. A bigger paging disk can only make the problem worse.
- c. Increase the degree of multiprogramming: No. This would only increase the number of processes and thus each process would get fewer frames. This would compound the thrashing problem.
- d. Decrease the degree of multiprogramming. Yes. This would mean fewer processes and in turn more frames per process, which could reduce thrashing.
- e. Install more main memory. Yes. More memory will reduce thrashing.
- f. Install a faster hard disk or multiple controllers with multiple hard disks. No. Thrashing isn't caused of disk speed, it's caused by memory size/availability.
- g. Add prepaging to the page-fetch algorithms. Yes. This technique loads all pages of a process's working set into memory before the process is restarted. Thus it can reduce thrashing.
- h. Increase the page size. No. Refer to the same answer in part b.