

**HW7**

**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
- b. FIFO replacement
- c. Optimal replacement
- d. Second-chance replacement



Suffers from Belady’s anomaly	Does not suffer from Belady’s anomaly
FIFO replacement	Optimal replacement
Second Chance replacement	LRU replacement

**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.**

- a. LRU replacement
- b. FIFO replacement
- c. Optimal replacement

Table 2: page replacements			
	LRU	FIFO	Optimal
1 Frame	20	20	20
2 Frames	18	18	15
3 Frames	15	16	11
4 Frames	10	14	8
5 Frames	8	10	7
6 Frames	7	10	7
7 Frames	7	7	7

**9.27 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, indicate whether it will (or is likely to) improve CPU utilization. Explain your answers.**

**a. Install a faster CPU.**

This is unlikely to improve CPU utilization because it will only make processes run faster, in return leaving the CPU is spent more time doing nothing while stalling for I/O.

**b. Install a bigger paging disk.**

This will also not help utilizing the CPU because it does not directly decrease the amount of time performing I/O requests or paging.

**c. Increase the degree of multiprogramming.**

No this will not improve CPU utilization because running twice as many programs does not decrease the amount time performing I/O requests or paging. It could actually causes twice as many and before.

**d. Decrease the degree of multiprogramming.**

This is likely to increase the CPU utilization. If we increase our degree of multiprogramming, we are over-allocating memory. If we run six processes, each of which is ten pages in size but actually uses only five pages, we have higher CPU utilization and throughput, with ten frames to spare.

**e. Install more main memory.**

This is also likely to increase the CPU utilization because it allows us to allocate more frames to memory and therefore less paging to occur.

**f. Install a faster hard disk or multiple controllers with multiple hard disks.**

Yes, this will improve CPU utilization as it widens the memory bottleneck meaning the CPU will get data more often and faster.

**g. Add prepaging to the page-fetch algorithms.**

Yes, because this will cause less paging later down the line.

**h. Increase the page size.**

Larger page sizes means that more frames can be allocated to processes and therefore less page faults to occur. Just like the examples in question 9.8