Trieu Huynh Ba Nguyen 000405980

1.

a. Disk scheduling algorithms, except for First-Come-First-Serve (FCFS), may not guarantee fairness in terms of waiting time for I/O requests. This is because some requests may constantly be postponed in favor of newer requests. For instance, in Shortest Seek Time First (SSTF), requests closest to the current position of the disk arm are selected first, while others may be continually deferred. This can lead to starvation of some requests, resulting in longer waiting times and decreased system performance.

b. A possible approach to ensure fairness in disk scheduling is to introduce aging or aging-based modifications to the existing algorithms. This involves setting an aging threshold, where any request that exceeds a predetermined time limit is moved to the top of the queue, and new requests are only added after all aged requests have been serviced. To facilitate this, a bit can be set to indicate that no new requests can be inserted ahead of the aged requests. For example, in SSTF, the remaining queue would need to be reorganized based on the position of the last aged request. By doing so, older requests receive priority over newer requests, thus reducing the likelihood of starvation.

c. Fairness is a crucial goal in a time-sharing system because it ensures that every user has equal access to system resources and that no user is unfairly prioritized over others. Fairness promotes equitable distribution of system resources, minimizes response time, and ensures that users are not discriminated against based on factors such as the time of their request or their position in the system. Without fairness, some users may experience longer response times, reduced system performance, and may be unable to complete their tasks, leading to dissatisfaction and poor user experience.

d. There are instances where the operating system may prioritize certain I/O requests over others, even if it means being unfair. For example, in a paging and swapping scenario, the system may prioritize disk accesses for memory allocation over user requests to maintain system stability. In addition, writing file system metadata may be given higher priority than user I/O requests to ensure the consistency and integrity of the file system. Also, in real-time systems, I/O requests from high-priority processes may be favored over those from lower-priority processes to meet strict timing requirements.

Trieu Huynh Ba Nguyen 000405980

2. The following table shows the access order of tracks and the total number of cylinders crossed:

Method	1	2	3	4	5	6	7	8	9	10	11	12	13	Total
FCFS	2150	2069	1212	2296	2800	544	1618	356	1523	4965	3681			13011
SSTF	2150	2069	2296	2800	3681	4965	1618	1523	1212	544	356			7586
SCAN	2150	2296	2800	3681	4965	4999	2069	1618	1523	1212	544	356		7492
LOOK	2150	2296	2800	3681	4965	2069	1618	1523	1212	544	356			7424
C-SCAN	2150	2296	2800	3681	4965	4999	0	356	544	1212	1523	1618	2069	9917
C-LOOK	2150	2296	2800	3681	4965	356	544	1212	1523	1618	2069			9137

SCAN and C-SCAN algorithms involve the movement of the disk head to a designated track, typically the last track numbered 4999, in the case of SCAN. In C-SCAN, however, the disk head only moves in a single direction, and after it reaches the last track, it returns to the first track, numbered 0, and continues scanning in the same direction.