Disk Scheduling

Carrying out disk accesses in the order they are received will not always produce optimal performance.

Seek time is the reason for differences in performance

For a single disk there will be a number of I/O requests

If requests are selected randomly, we will expect poor performance

Can use priority scheme

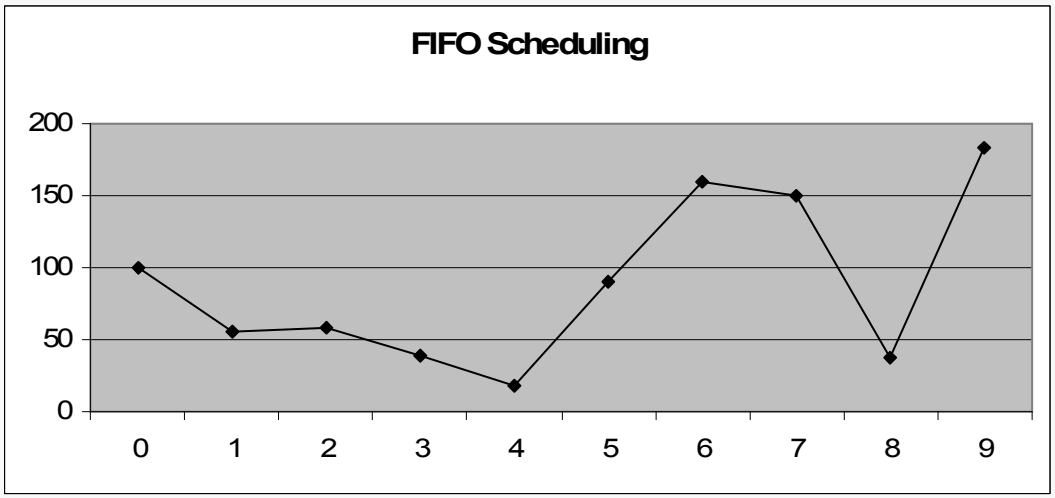
Can reduce average access time by sending requests to disk controller in certain order

FIFO Scheduling

First-in, first-out (FIFO)

- process request sequentially
- "fair" to all processes
- approaches random scheduling in performance if there are many processes

Request order: 55 58 39 18 90 160 150 38 184



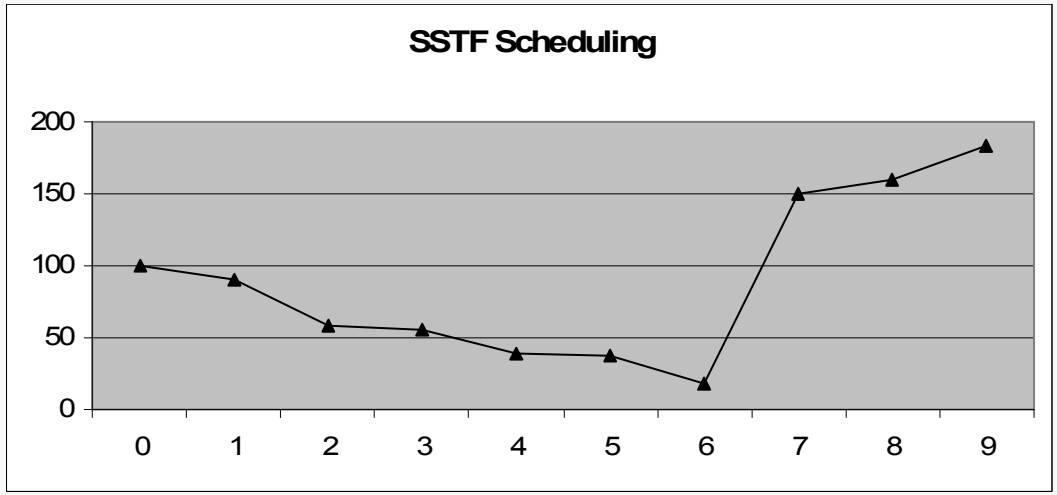
Total distance head moves: 498

SSTF Scheduling

SSTF: shortest seek (service) time first

- select the disk I/O request that requires the least movement of the disk arm from its current position
- guarantees minimum average seek time, but can lead to starvation

Request order:	55	58	39	18	90	160	150	38	184
Actual order:	90	58	55	39	38	18	150	160	184



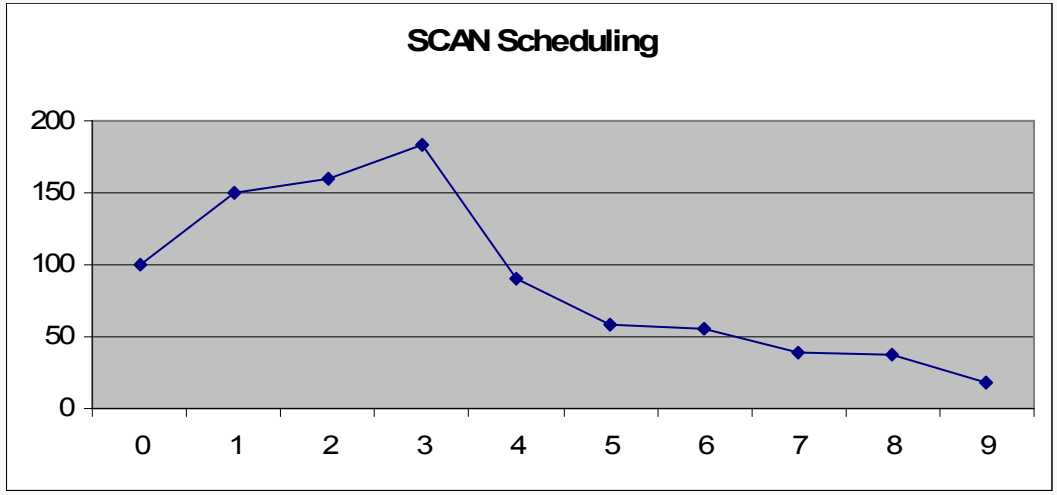
Total distance head moves: 248

SCAN Scheduling

SCAN: "elevator algorithm"

- arm moves in one direction only, satisfying all outstanding requests until it reaches the last track in that direction
- then direction is reversed

Request order:	55	58	39	18	90	160	150	38	184
Actual order:	150	160	184	90	58	55	39	38	18



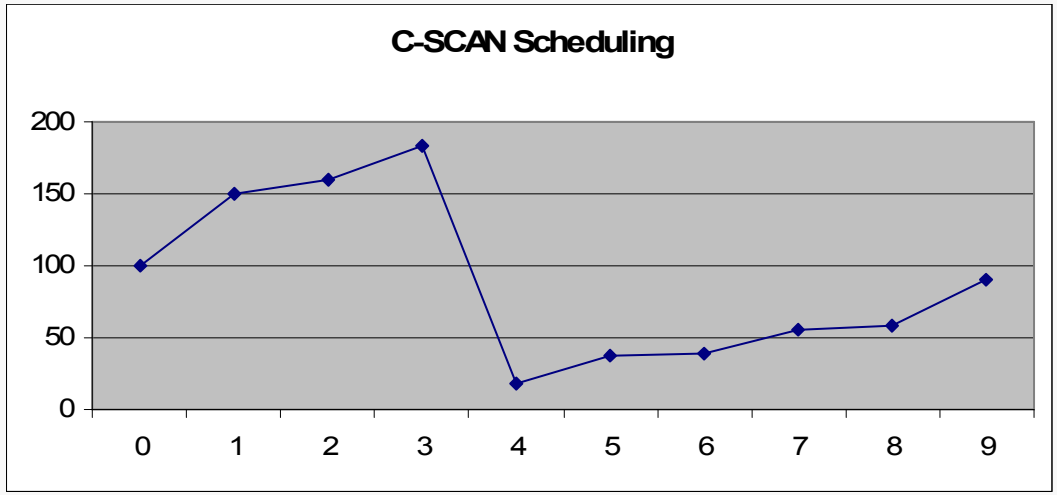
Total distance head moves: 250

C-SCAN Scheduling

C-SCAN:

- restricts scanning to one direction only
- when the last track has been visited in one direction, the arm is returned to the opposite end of the disk and the scan begins again
- more uniform waiting times
- "fairer" than SCAN

Request order:	55	58	39	18	90	160	150	38	184
Actual order:	150	160	184	18	38	39	55	58	90



Total distance head moves: 312

Other Variations

N-step-SCAN

- Segments the disk request queue into subqueues of length N
- Subqueues are processed one at a time, using SCAN
- New requests added to other queue when queue is processed

FSCAN

- Two queues
- One queue is empty for new requests

Comparison

(a) FIFO (starting at track 100)		(b) SSTF (starting at track 100)		(c) SCAN (starting at track 100, in the direction of increasing track number)		(d) C-SCAN (starting at track 100, in the direction of increasing track number)	
Next track accessed	**Number of tracks traversed**	**Next track accessed**	**Number of tracks traversed**	**Next track accessed**	**Number of tracks traversed**	**Next track accessed**	**Number of tracks traversed**
55	45	90	10	150	50	150	50
58	3	58	32	160	10	160	10
39	19	55	3	184	24	184	24
18	21	39	16	90	94	18	166
90	72	38	1	58	32	38	20
160	70	18	20	55	3	39	1
150	10	150	132	39	16	55	16
38	112	160	10	38	1	58	3
184	146	184	24	18	20	90	32
Average seek length	55.3	**Average seek length**	27.5	**Average seek length**	27.8	**Average seek length**	35.8

Fallacy

Operating systems are the best place to manage the scheduling of disk accesses.

Problem: high-level interfaces like ATA and SCSI provide the OS with logical block addresses, not physical disk addresses.

Host-Ordered vs Drive-Ordered

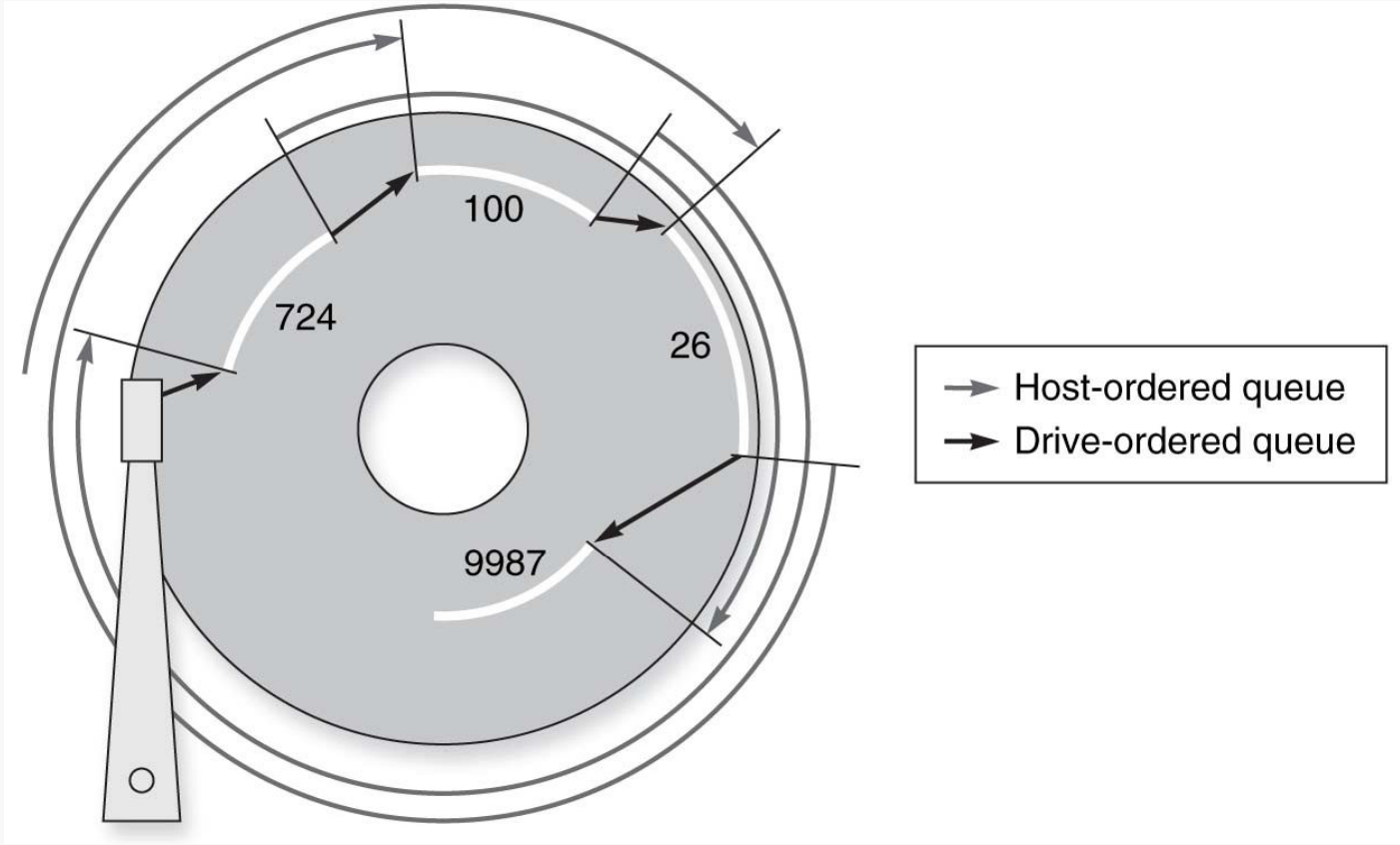


FIGURE 6.19 Example showing OS versus disk schedule accesses, labeled host-ordered versus drive-ordered. The former takes three revolutions to complete the four reads, while the latter completes them in just three-fourths of a revolution (from Anderson [2003]).