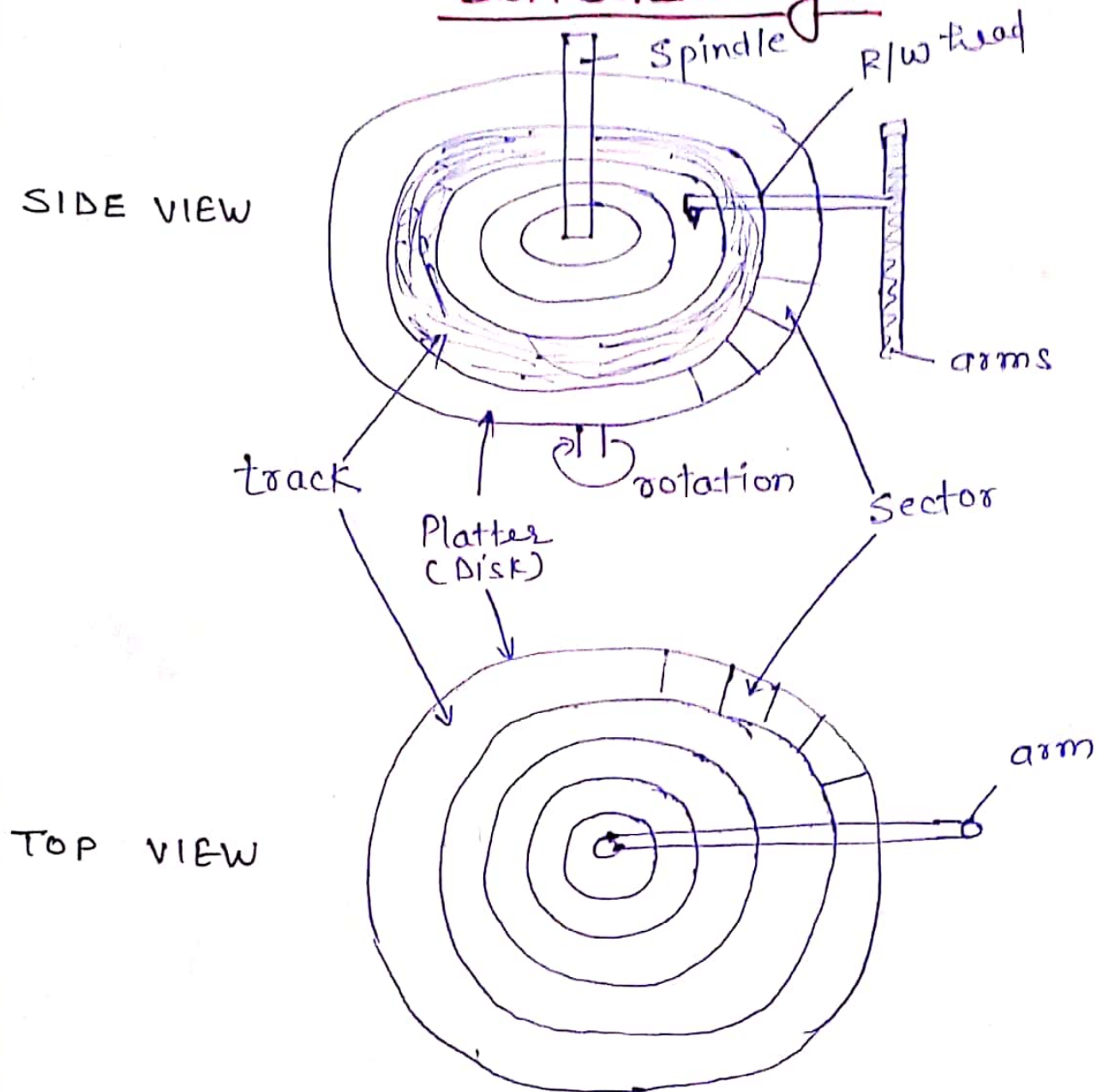


Disk Scheduling.



- Each platter (Disk) has a flat circular shape like a CD or DVD.
- Generally diameter ranges from 1.8 to 5.25 inches.
- Both surface are covered by a magnetic materials.
- R/w head is attached to disk arm.
- Surface of disk is divided into circular tracks.
- tracks are further divided into ~~circular~~ sectors.

(D.1)

Seek time - time required to move the R/W head on the desired track.

Disk Scheduling - In case of multiple I/O request disk scheduling algo must decide which required must be executed first.

Rotational Latency - It is the time spent waiting for the target sector to appear under the read and write head.

Numericals.

1. Seek time :

$$T_s = m \times n + s$$

where

T_s = seek time

n = No. of track traversed.

m = Constant that depends on the disk drive.

s = startup time.

2. Transfer Time : The transfer time to or from the disk depends on the rotation speed of disk in the following fashion :

(D.2)

$$T = \frac{b}{rN}$$

where

T = Transfer time

b = Number of bytes to be transferred.

r = Rotation speed in revolutions per second.

N = No. of bytes on the track.

The total average access time can be expressed as

$$T_a = T_s + \frac{1}{2r} + \frac{b}{rN}$$

where T_s = Average seek time.

3. Total capacity of disk :

= No. of cylinders $\times$ No. of heads $\times$
 No. of sectors per track $\times$ No. of
 bytes per sectors.

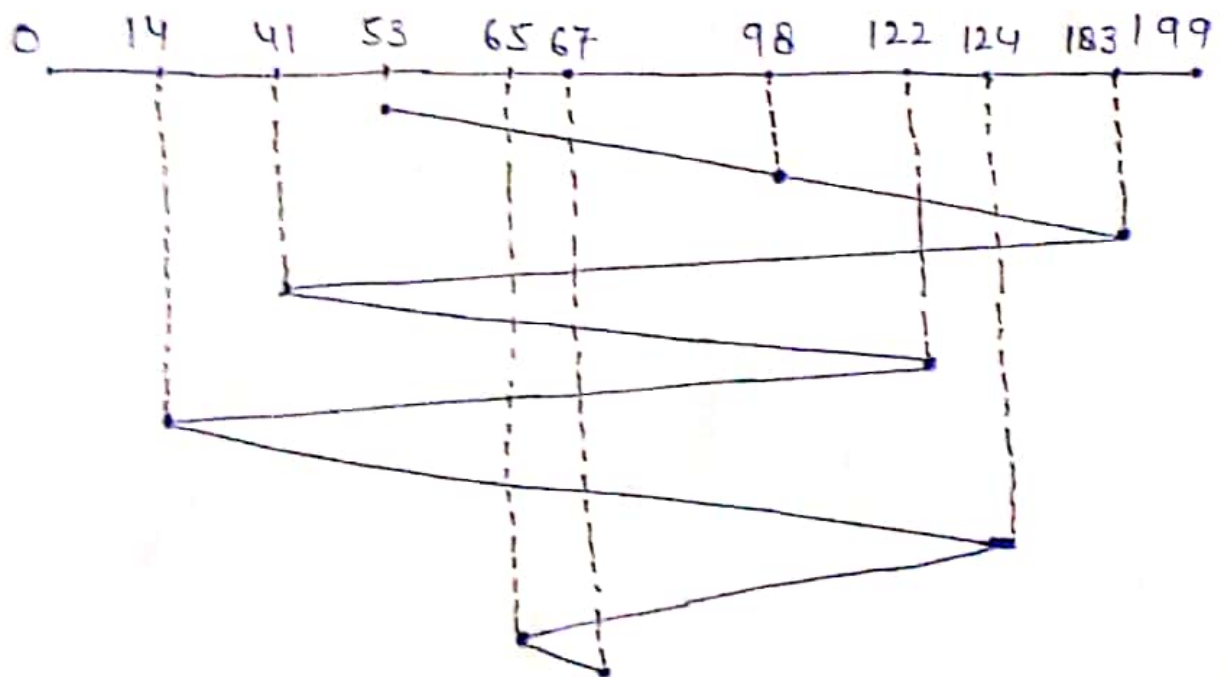
(643)

Disk Scheduling Algorithm

(1) First Come First Serve (FCFS) :

- Simplest disk scheduling algo.
- Request are entertained in the order they arrive (enter disk queue.)

98, 183, 41, 122, 14, 124, 65, 67
R/w Head at 53.



$$\begin{aligned} 98 - 53 &= 45 \\ 183 - 98 &= 85 \\ 183 - 41 &= 142 \\ 122 - 14 &= 108 \\ 124 - 14 &= 110 \end{aligned}$$

$$\begin{aligned} 124 - 65 &= 59 \\ 67 - 65 &= 02 \\ \hline 632 \end{aligned}$$

Total 632 moves.
either called as total
seek movements or track
movements. (D.4)

Advantage :

- Easy to understand and implement.
- No starvation. (may suffer from convoy effect)
- Can be used with less load.

Disadvantage :

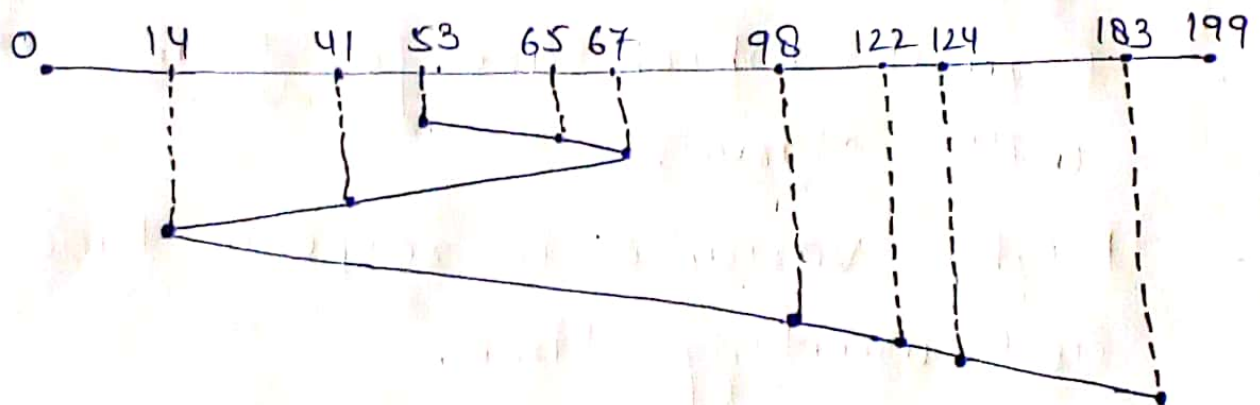
- Require more seek time.
- More waiting time and response time.
- Inefficient.

2. Shortest Seek Time First (SSTF) :

- Serve the request which is closest to the current position of R/W head.
- Tie is broken in the direction of head movement.

98, 183, 41, 122, 14, 124, 65, 67

R/W head at 53. ↑



$$65 - 53 = 12$$

$$67 - 65 = 2$$

$$67 - 41 = 26$$

$$41 - 14 = 27$$

$$98 - 14 = 84$$

$$122 - 98 = 24$$

$$124 - 122 = 2$$

$$183 - 124 = 59$$

$$\hline 236$$

Total seek movement is 236.

(D.6)

Advantages :

- Very efficient in seek moves.
- less average response time and waiting time.
- Increased throughput.

Disadvantages :

- Overhead to find out the closest request.
- Request which are far from head will starve.
- High variance in response time and waiting time.

3. SCAN :

- Head start at one end of the disk and moves towards the other end,
- Serving requests in between, and reach other end and then direction of head is reversed and process continues.
- Head continuously scan back and forth across the disk.
- Sometimes called Elevator algo.
- R/W head moves like sweeper clean the platform.

Advantages :

- Simple, easy to understand and implement.
- No starvation. (but bounded waiting).
- Low variance and average waiting time.

Disadvantages :

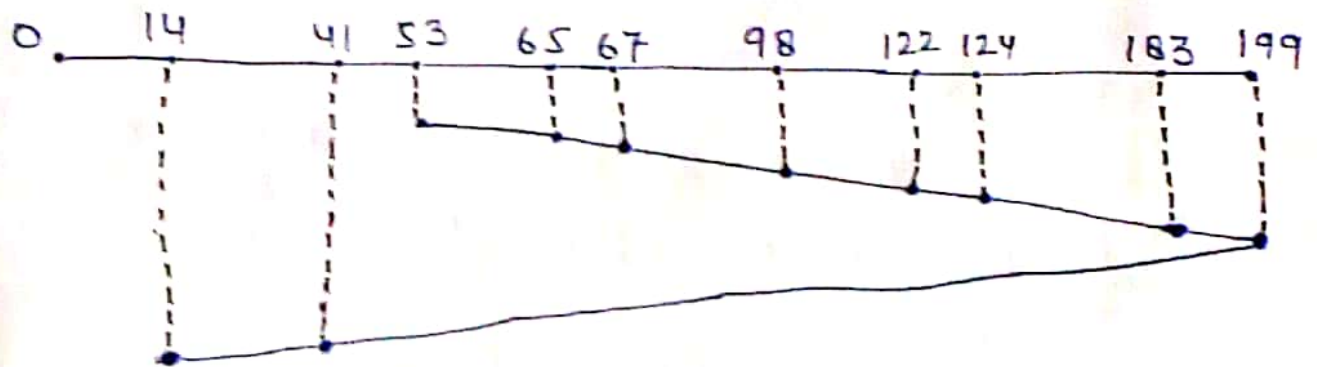
- Long waiting time for the locations just visited by head.

(D.8)

- Unnecessary moves till the end of the disk, even if there is no request.

98, 183, 41, 122, 14, 124, 65, 67

R/w head is at 53↑



We don't have request for 199, but in the algo. head needs to move till the end of disk.

$$\begin{aligned} & (65 - 53) + \\ & (67 - 65) + \\ & (98 - 67) + \\ & (122 - 98) + \\ & (124 - 122) + \\ & (183 - 124) + \\ & (199 - 183) + \\ & (199 - 41) + \\ & (41 - 14) = 331 \end{aligned}$$

Total seek movement is 331.

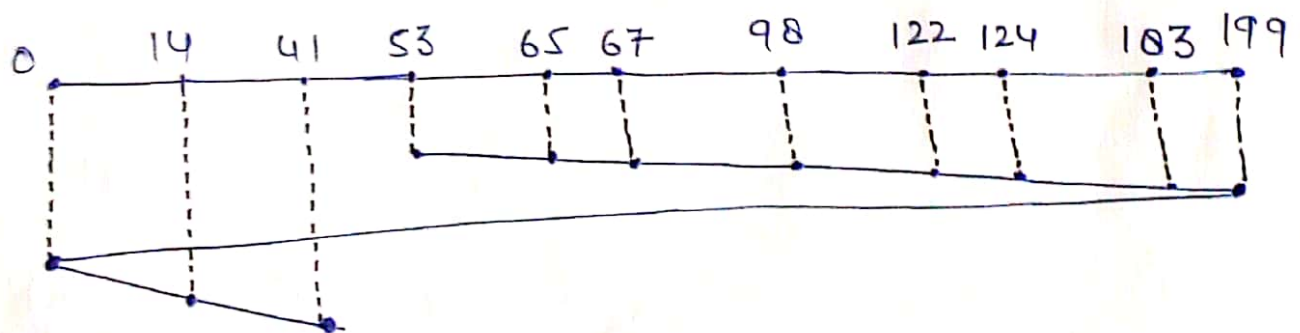
(D.9)

4. Circular Scan (C-SCAN) :

- Head start at one end of the disk and moves towards the other end, servicing requests in between, and reach other end and then direction of head is reversed and head reaches first end without satisfying any request.

98, 183, 41, 122, 14, 124, ~~65~~, 67

R/w head at 53↑



$$\begin{aligned} & (65-53) + (67-65) + (98-67) + (122-98) \\ & + (124-122) + (183-124) + (193-183) + (199-0) \\ & + (14-0) + (41-14) = 386 \end{aligned}$$

Total seek movement is 386.

Advantages :

- provides uniform waiting time.
- Better response time.

Disadvanges :

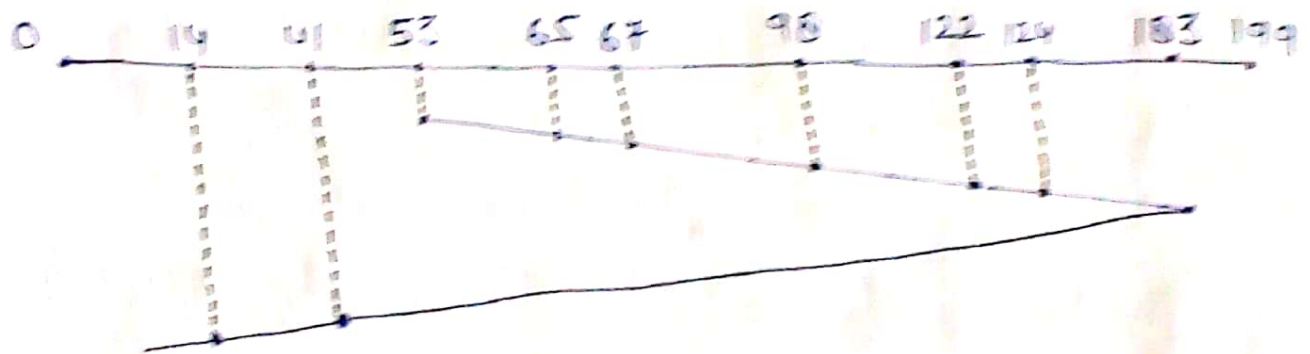
- More seek movements compared to simple SCAN.

5. LOOK :

- It is same as SCAN algo, but instead of going till last track, we go till last request, and then change direction.

98, 183, 41, 122, 14, 124, 65, 67

R/W Head is at 53↑



$$\begin{aligned} & (65-53) + (67-65) + (98-67) + (122-98) + \\ & (124-122) + (183-124) + (183-41) + (41-14) \\ & = 299 \end{aligned}$$

Total seek movement is 299.

Advantages :

- Better performance compared to SCAN.
- should be used in case of less load.

Disadvantages :

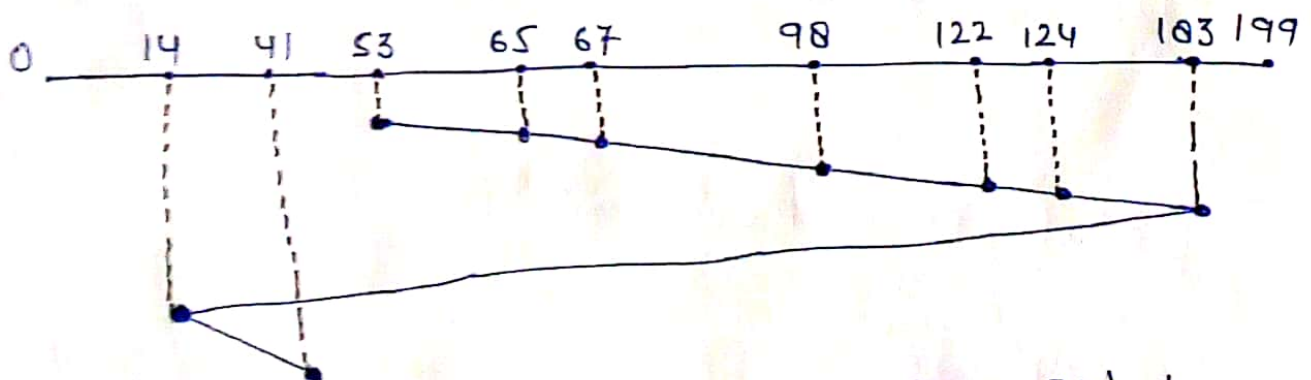
- Overhead to find the last request.
- should not be used in case of more load.

6. C-LOOK :

- It takes the advantage of both C-SCAN and LOOK algo.
- Will satisfy the request only in one direction.
- Will go till last request and return but not till last track.

98, 183, 41, 122, 14, 124, 65, 67

R/W Head at 53 ↑



$$\begin{aligned} & (65-53) + (67-65) + (98-67) + (122-98) + \\ & (124-122) + (183-124) + (183-14) + (41-14) \\ & = 326 \end{aligned}$$

Total seek movement is 326.

(D.13)

Advantages :

- More uniform waiting time to LOOK.
- More efficient (less moves) compared to C-SCAN.

Disadvantages :

- More overhead in calculations.
- Should not be used in case of more load.