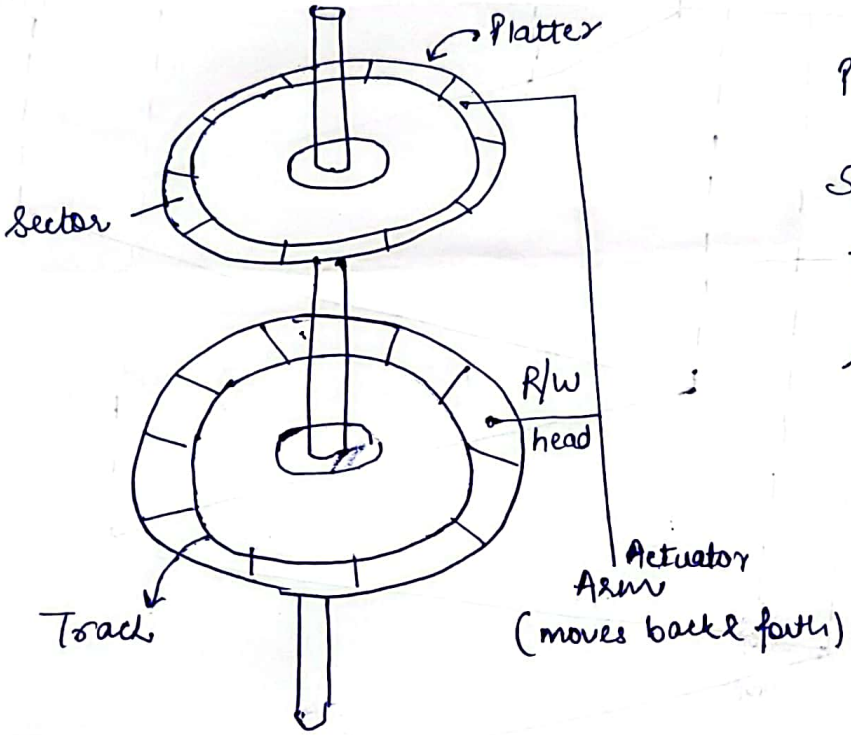
→ Disk Scheduling Algorithms

- FCFS
- SSTF
- SCAN
- C-SCAN

→ Direct Memory Access

* Goal of the disk scheduling algorithms is to minimize the seek time.
* Seek Time: Time Taken to reach upto desired track
* Hard disk is a collection of platters



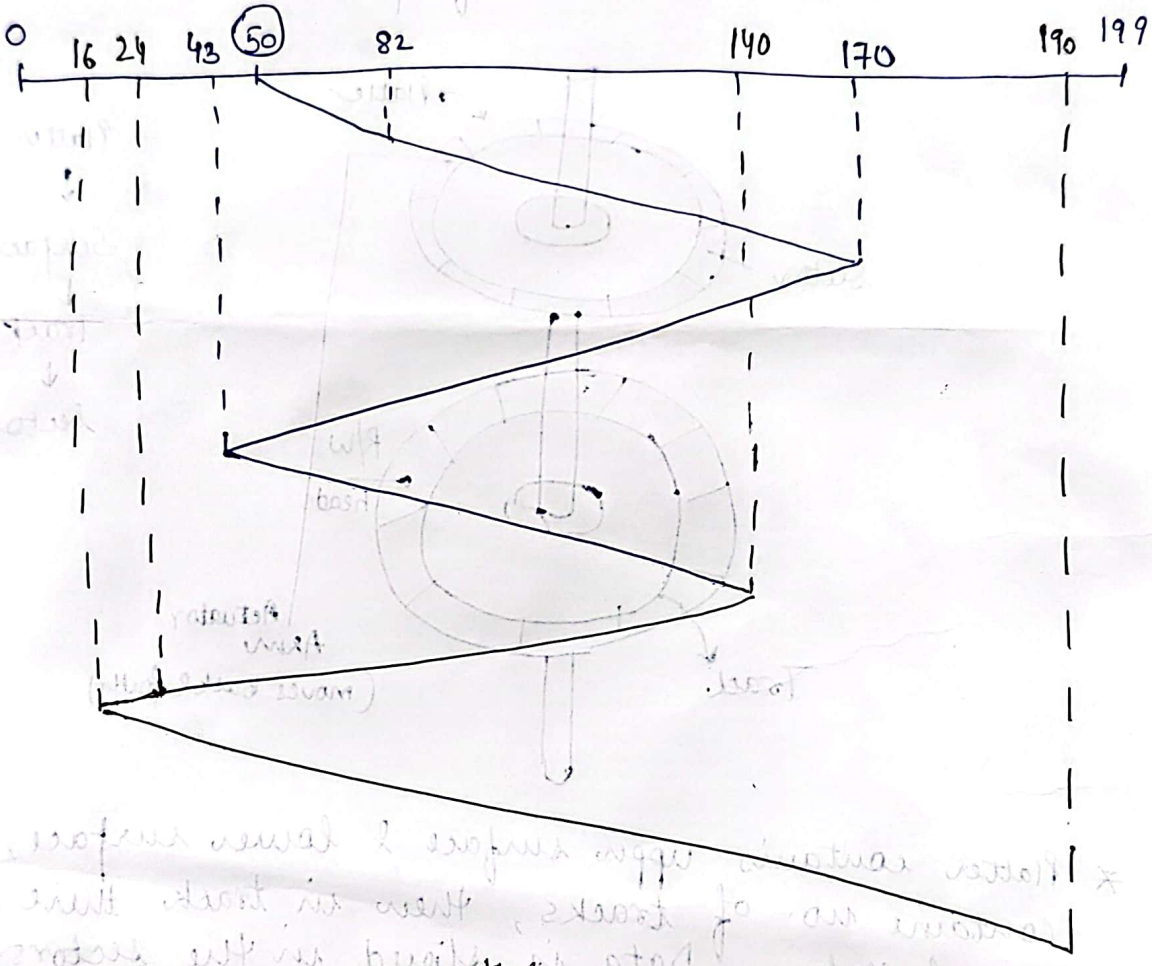
Platter
↓
Surface
↓
Track
↓
Sector

* Platter contains upper surface & lower surface, a surface contains no. of tracks, then in track there are no. of sectors. Data is stored in the sectors.
* We have to move R/W head to the desired track. The time taken to move the R/W head on to the desired track is called as SEEK TIME.

→ First come first serve

Ques. A disk contains 200 tracks (0-199)
Request queue contains track no. 82, 170, 43, 140, 24, 16, 190 respectively.
Current position of R/W head = 50
Cal total no. of tracks movement by RW.

* Seek time: No. of movements by R/W head.



Total no. of track movements:

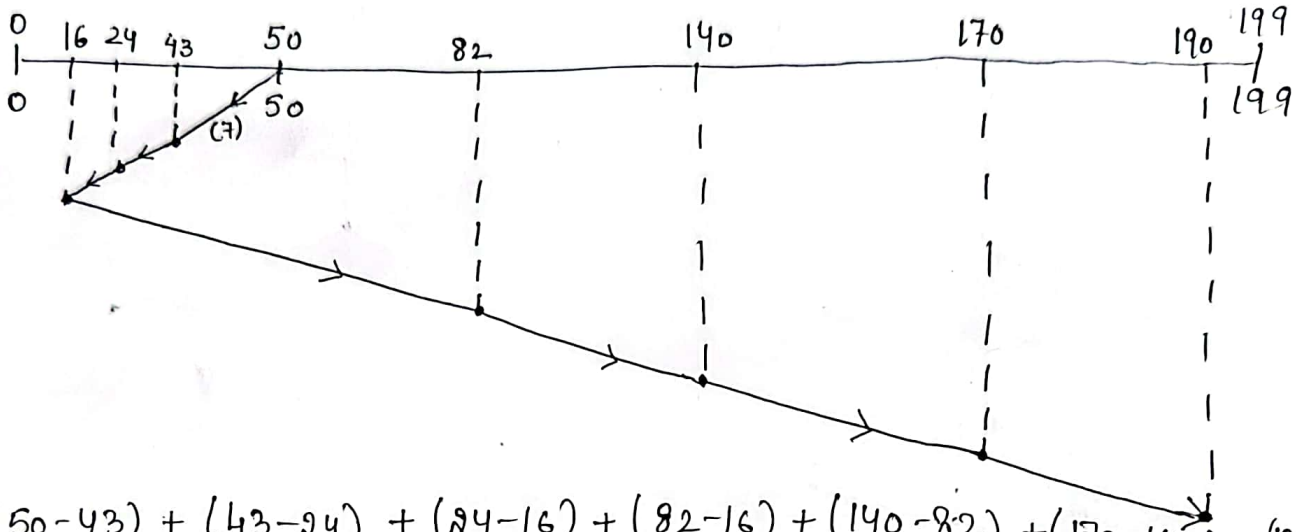
(I) (82-50) + (170-82) + (170-43) + (140-43) + (140-24) + (24-16) + (190-16).

(II) (170-50) + (170-43) + (140-43) + (140-16) + (190-16)
= 642

* There is no problem of starvation in FCFS but direction change is allot ∴ seek time value is more.

ortest Seek Time first (SSTF)

- A disk contains 200 tracks (0-199), Request queue contains track no. 82, 170, 43, 140, 24, 16, 190 respectively.
- Current position of R/W head = 50. Calculate total no. of track movement by R/W head using Shortest Seek time first.
- If R/W head takes 1 ns to move from one track to another, the total time taken 208 x 1ns = 208 ns?



I: (50-43) + (43-24) + (24-16) + (82-16) + (140-82) + (170-140) + (190-170)

II: (Direction Wise)

(50-16) + (190-16) = 208

Advantages:

(i) It tries to give optimal result (shortest seek time)

(ii) Response time is good.

Disadvantages:

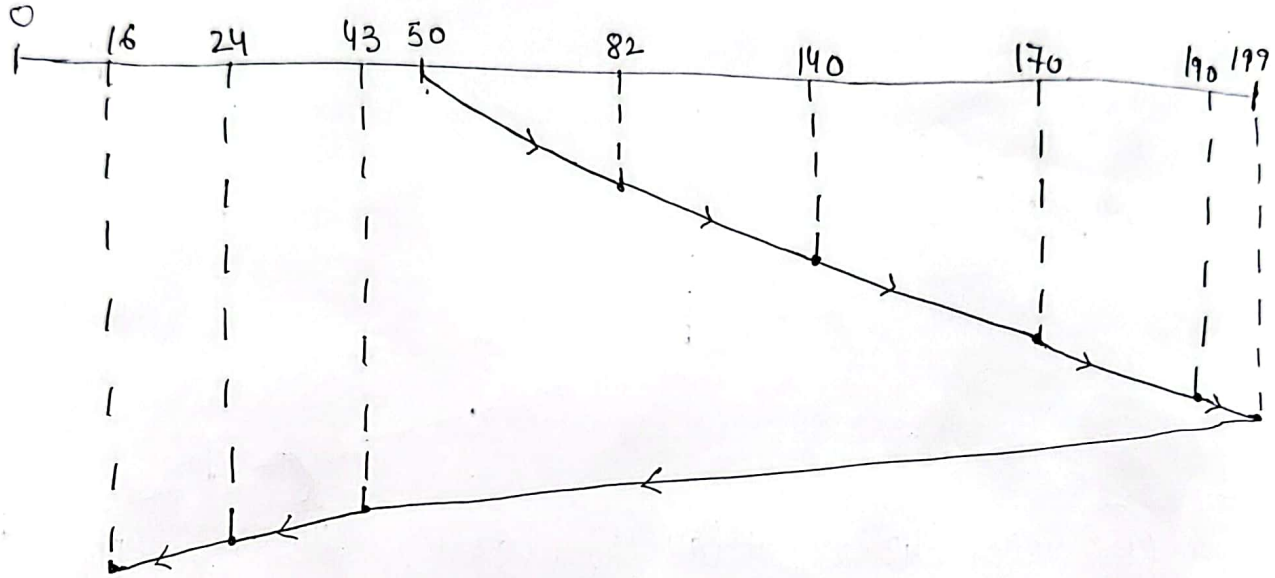
(i) Starvation (one of the request which came at first has to wait for a longer time if it is seek time is longer)

(ii) Overhead is generated (means we have to find which is the nearest request next every time).

→ SCAN Algorithm / Elevator Algorithm

Ques: A disk contains 200 tracks (0-199). Request queue con... track no. 82, 170, 43, 140, 24, 16, 190 respectively. Current position of R/W head = 50.

- Calculate total no of track movement by R/W head using SCAN ?
- if R/W head takes 1 ns to move from one track to another then total time taken $332 \times 1ns = 332ns$.
- Direction towards the large value.



* Here the algorithm will move towards one direction first till the end but when the direction is changed then it will move till the last request not till the end.

I :

I : (199-50) + (199-16) = 332

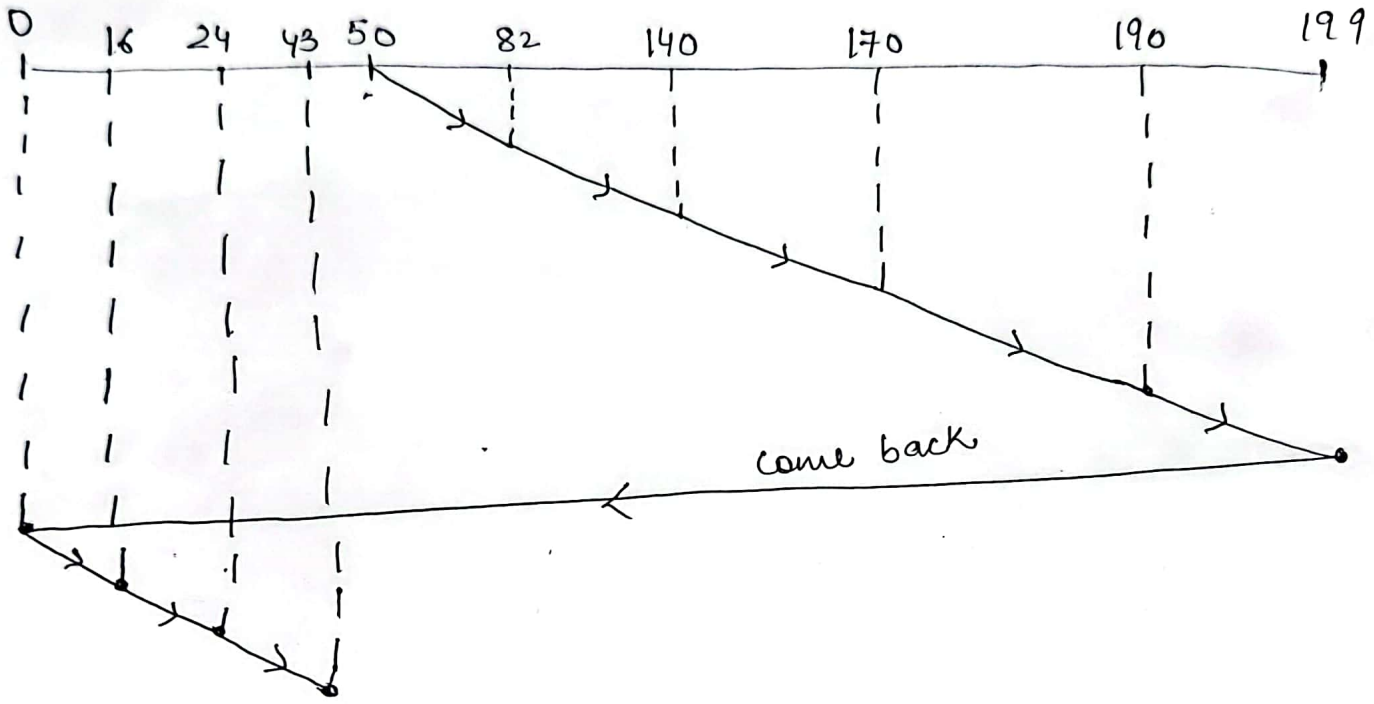
Disadvantage

1) Once the direction is changed and in mean time we receive the request for the previous direction, then it cannot go back, it will complete the ongoing direction first.

C-SCAN / Circular-SCAN Algorithm

Ques: A disk contains 200 tracks (0-199), Request queue contains track no. 82, 170, 43, 140, 24, 16, 190 respectively.

- Current position of R/W head = 50.
- Cal. total no. of tracks movement by R/W head using C-SCAN?
- Direction is towards large value.



I:

II: (199-50) + (199-0) + (43-0) = 391