LOOK Disk Scheduling Algorithm

The LOOK Disk Scheduling Algorithm

The LOOK Disk Scheduling Algorithm is the advanced version of the SCAN (elevator) disk scheduling algorithm which gives slightly better seek time than any other algorithm in the hierarchy (FCFS->SRTF->SCAN->C-SCAN->LOOK). It is used to **reduce the amount of time it takes to access data on a hard disk drive by minimizing the seek time between read/write operations**. The LOOK algorithm operates by **scanning the disk in a specific direction**, but instead of going all the way to the end of the disk before reversing direction like the SCAN algorithm, it **reverses direction as soon as it reaches the last request in the current direction**.

The LOOK Disk Scheduling Algorithm

- LOOK serves requests in one direction until none remain.
- It avoids going to disk ends unless needed.
- This reduces unnecessary movement.

Importance of The LOOK Disk Scheduling Algorithm

The LOOK disk scheduling algorithm is important because it optimizes disk performance by **reducing unnecessary movements of the disk arm**. By serving requests in one direction until no more remain, it **saves time and improves efficiency**. Unlike other algorithms, it **skips over empty areas**, which reduces delays and speeds up processing. This makes LOOK a practical choice for systems that need fast and efficient disk operations.

Steps involved in The LOOK Disk Scheduling Algorithm

1. Determine the initial direction of disk head movement.
2. Sort the pending disk requests in the order in which they will be serviced.
3. Scan the disk in the chosen direction, servicing requests as they are encountered
4. When the last request in the current direction has been serviced, reverse the direction and continue scanning until all requests have been serviced.

Advantages of
The LOOK Disk Scheduling Algorithm

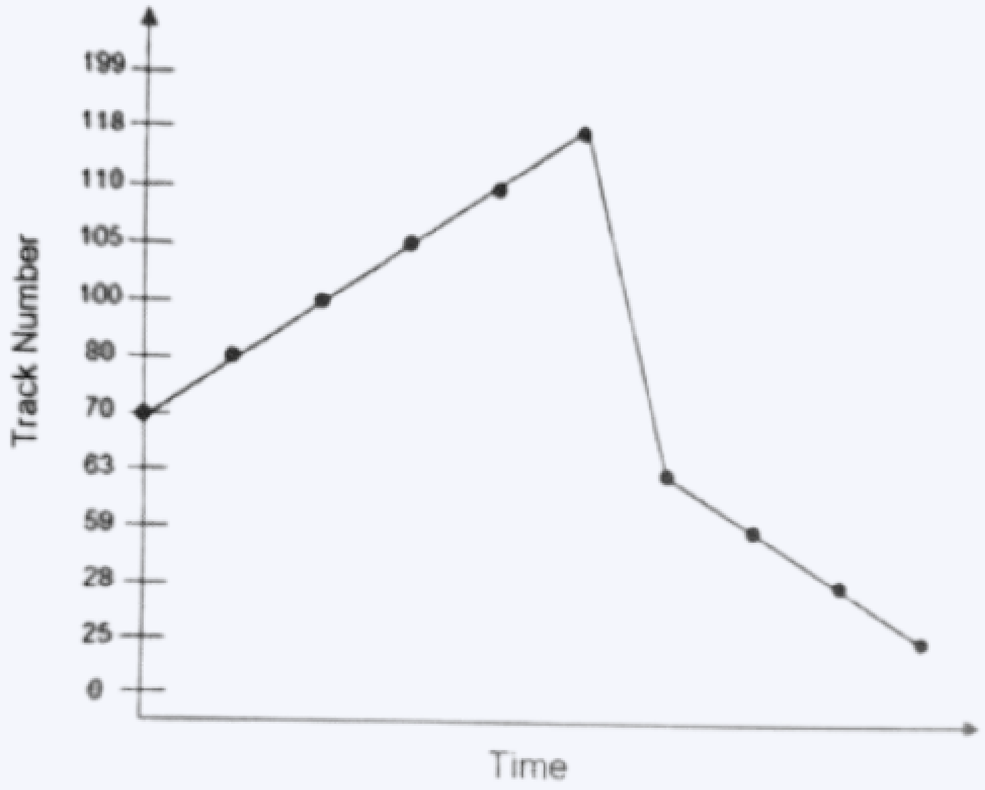
- Reduces unnecessary disk movement.
- More efficient than SCAN in some cases.
- Handles requests faster by skipping empty areas.

Disadvantages of The LOOK Disk Scheduling Algorithm

- May cause starvation for distant requests.
- Performance depends on request distribution.
- Can be less predictable than other algorithms.

The LOOK Disk Scheduling Algorithm: Line Graph and Calculation

The LOOK Disk Scheduling Algorithm Example #1



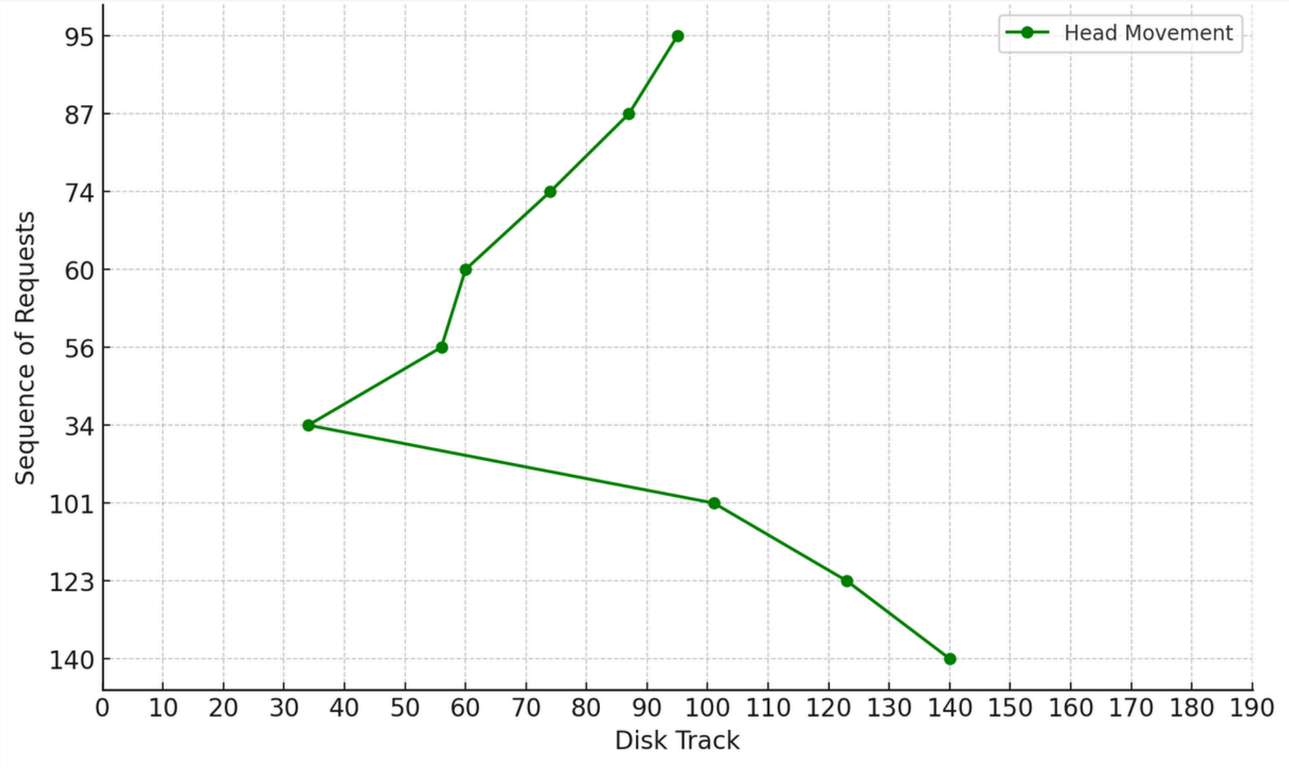
The total number of head movements is:

= (80-70) + (100-80) + (105-100) + (110-105) + (118-110) + (118-63) + (63-59) + (59-28) + (28-25)

= 10 + 20 + 5 + 5 + 8 + 55 + 4 + 31 + 3

= **141 tracks**

The LOOK Disk Scheduling Algorithm Example #2



The total number of head movements is:

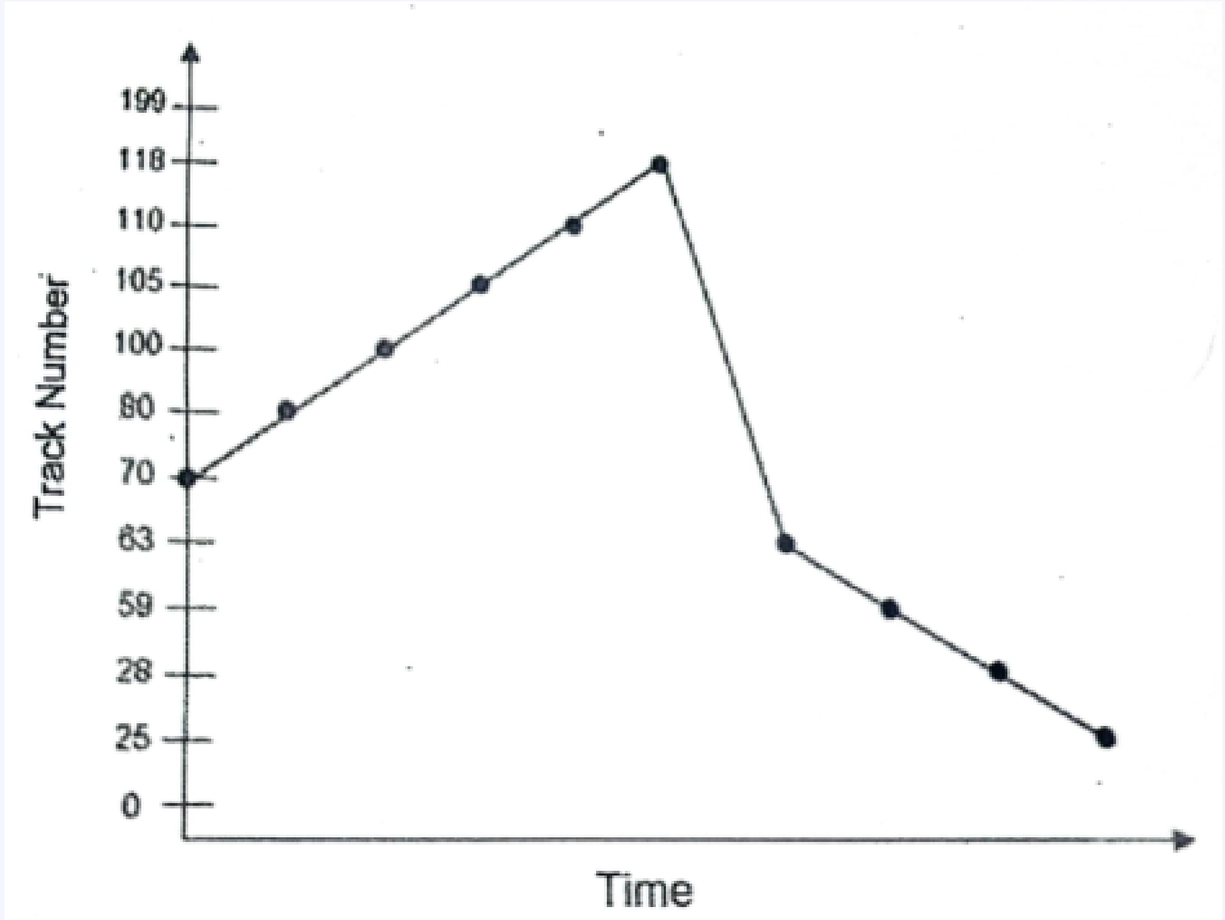
=(95 - 87) + (87 - 74) + (74 - 60) + (60 - 56) + (56 - 34) + (101 - 34) + (123 - 101) + (140 - 123)

= 8 + 13 + 14 + 4 + 22 + 67 + 22 + 17

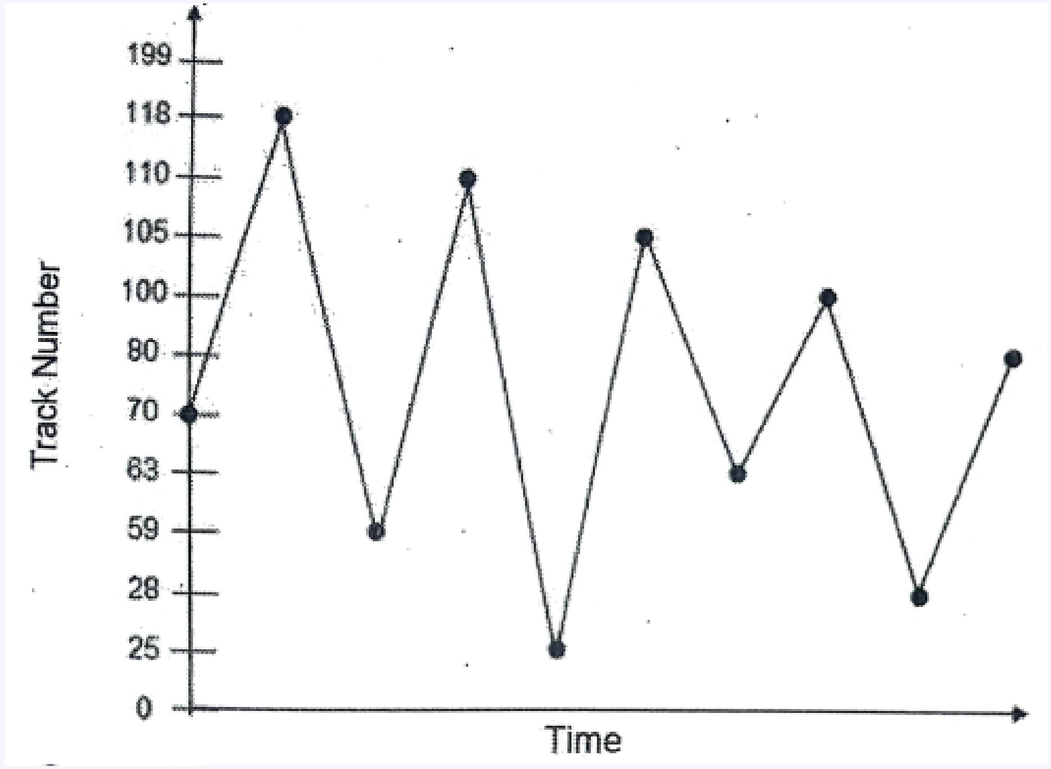
= 167 tracks

Comparisons of Disk Scheduling Algorithms

LOOK SCHEDULING ALGORITHM



FCFS SCHEDULING ALGORITHM



LOOK SCHEDULING ALGORITHM

The total number of head movements for LOOK Disk Scheduling Algorithm

= (80 - 70) + (100 - 80) + (105 - 100) + (110 - 105) + (118 - 110) + (118 - 63) + (63 - 59) + (59 - 28) + (28 - 25)

= 10 + 20 + 5 + 5 + 8 + 55 + 44 + 31 + 3

= **141 tracks**

FCFS SCHEDULING ALGORITHM

The total number of head movements for FCFS Disk Scheduling Algorithm

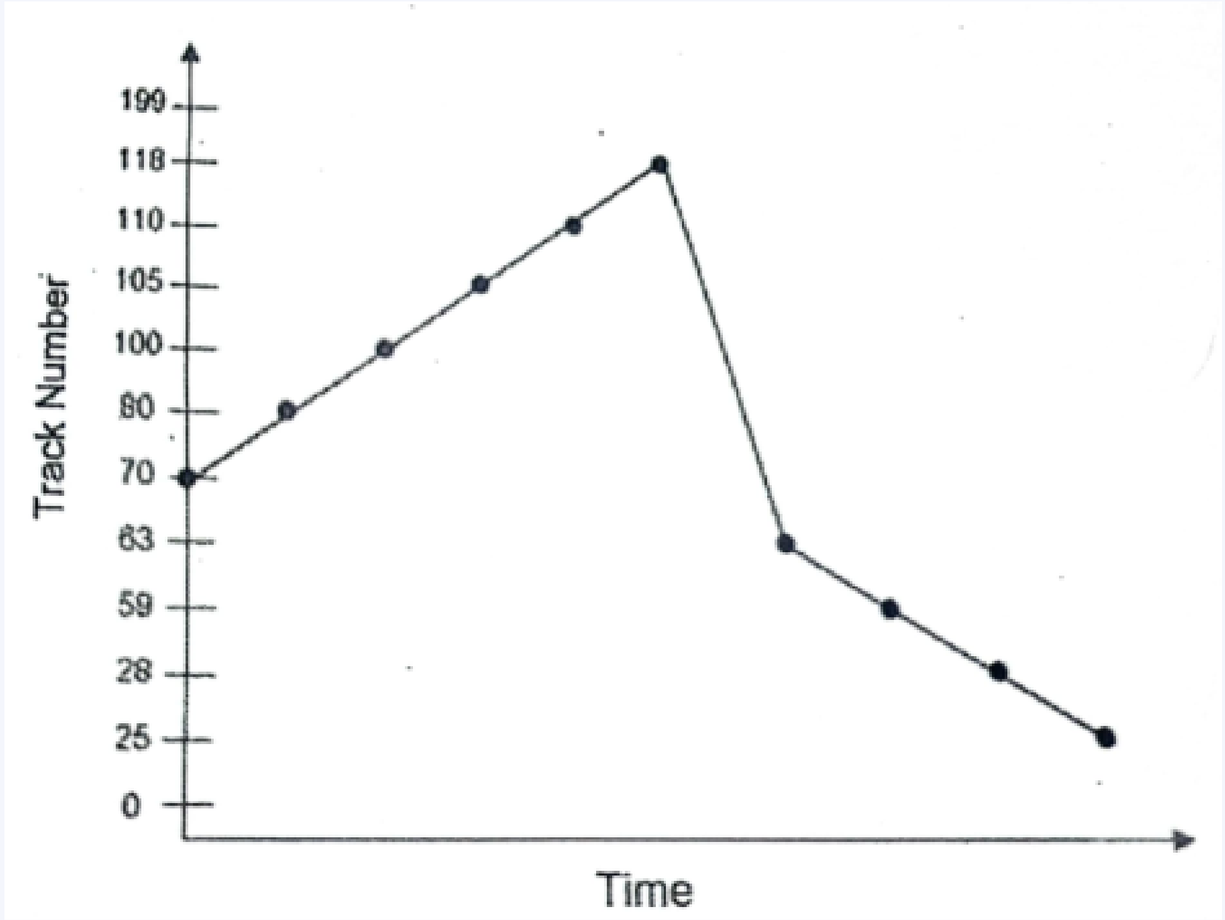
= (118 - 70) + (118 - 59) + (110 - 59) + (110 - 25) + (105 - 25) + (105 - 63) + (100 - 63) + (100 - 28) + (80 - 28)

= 48 + 59 + 51 + 85 + 80 + 42 + 37 + 72 + 52

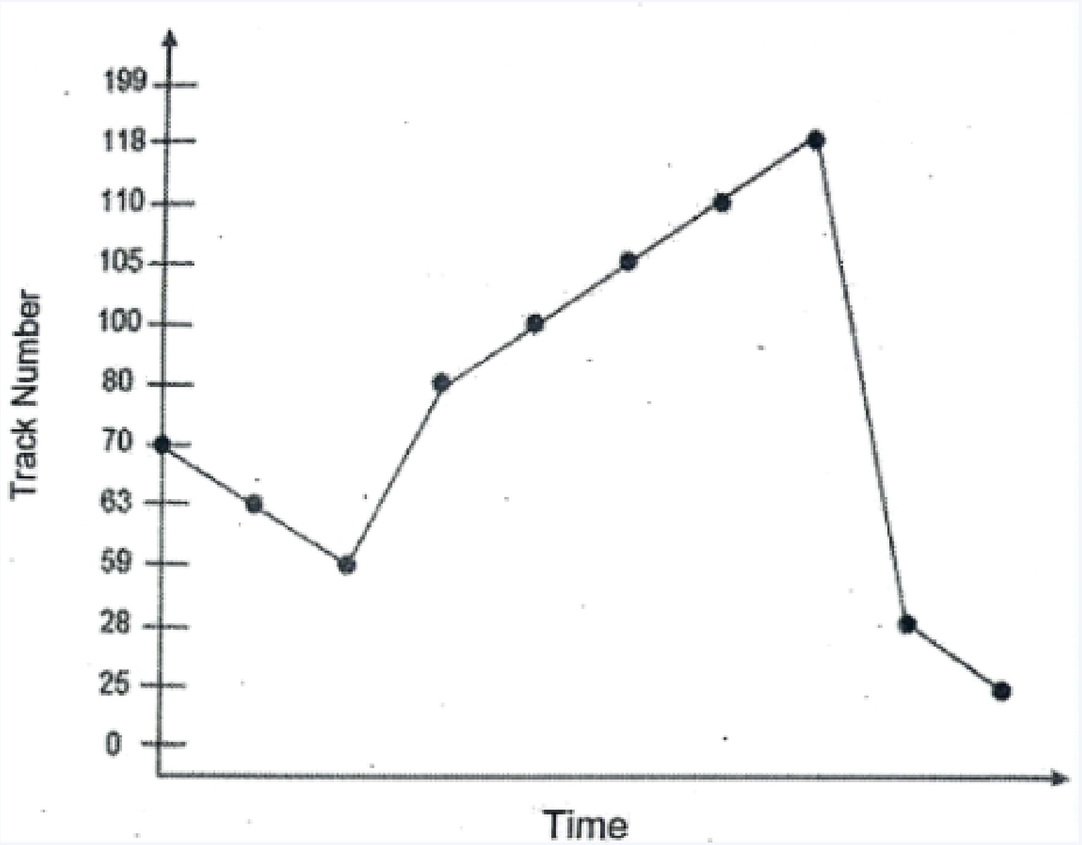
=526 tracks

Comparisons of Disk Scheduling Algorithms

LOOK SCHEDULING ALGORITHM



SSTF SCHEDULING ALGORITHM



LOOK SCHEDULING ALGORITHM

The total number of head movements for LOOK Disk Scheduling Algorithm

= (80 - 70) + (100 - 80) + (105 - 100) + (110 - 105) + (118 - 110) + (118 - 63) + (63 - 59) + (59 - 28) + (28 - 25)

= 10 + 20 + 5 + 5 + 8 + 55 + 44 + 31 + 3

= **141 tracks**

SSTF SCHEDULING ALGORITHM

The total number of head movements for SSTF Disk Scheduling Algorithm

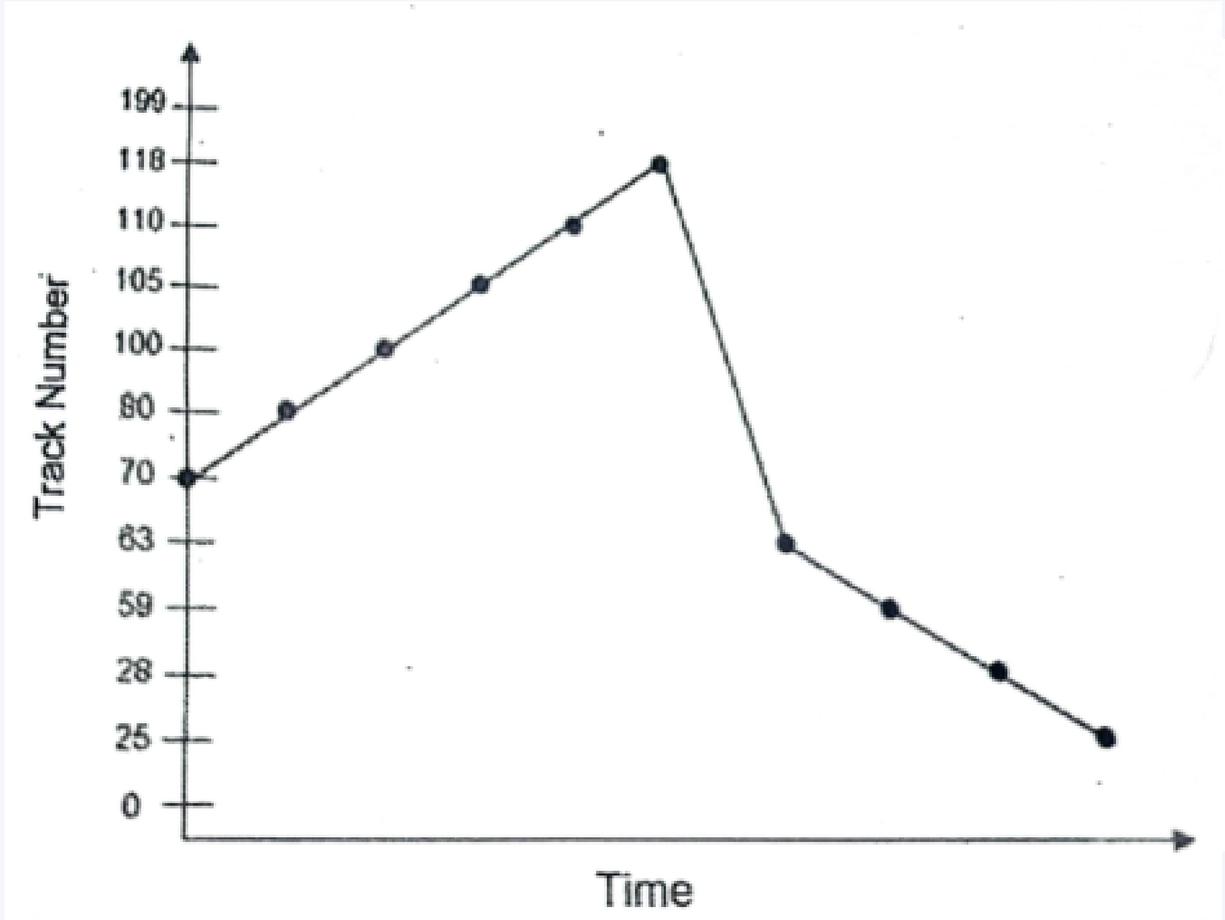
= (70 - 63) + (63 - 59) + (80 - 59) + (100 - 80) + (105 - 100) + (110 - 105) + (116 - 110) + (118 - 28) + (28 - 25)

= 7 + 4 + 21 + 20 + 5 + 5 + 8 + 90 + 3

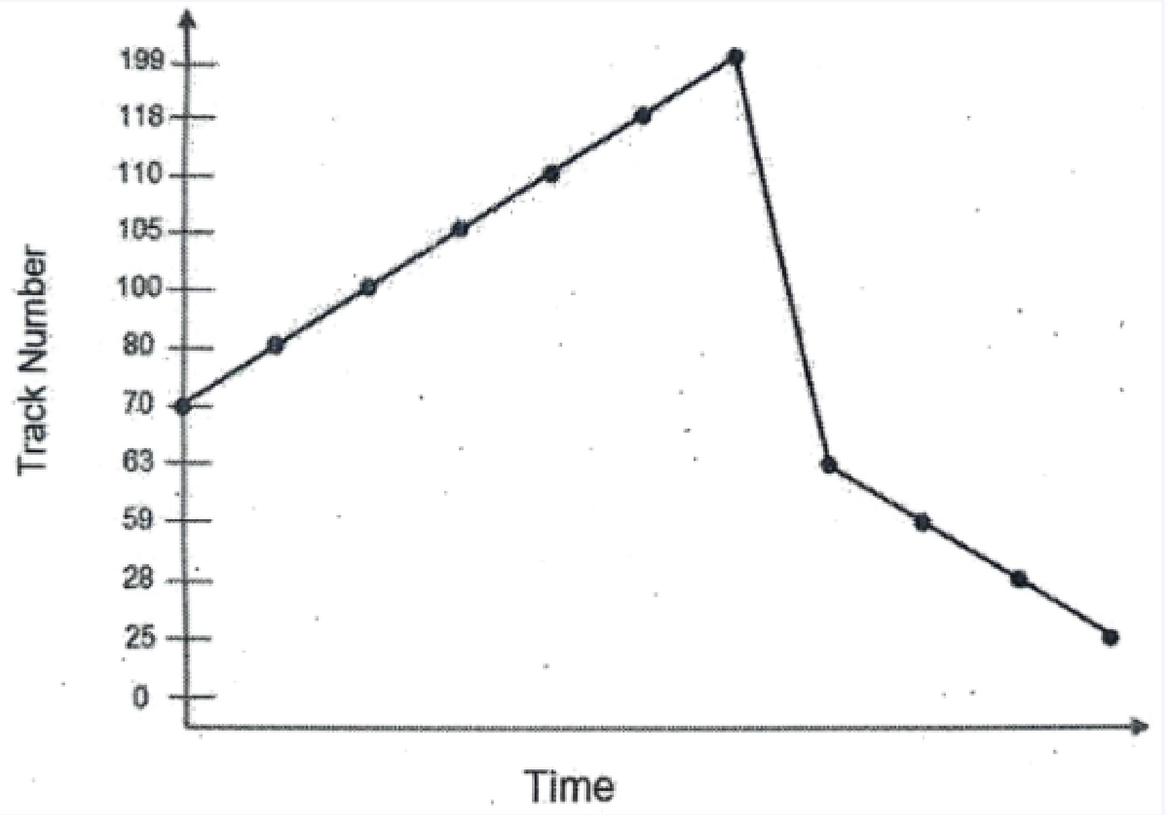
=163 tracks

Comparisons of Disk Scheduling Algorithms

LOOK SCHEDULING ALGORITHM



SCAN SCHEDULING ALGORITHM



LOOK SCHEDULING ALGORITHM

The total number of head movements for LOOK Disk Scheduling Algorithm

= (80 - 70) + (100 - 80) + (105 - 100) + (110 - 105) + (118 - 110) + (118 - 63) + (63 - 59) + (59 - 28) + (28 - 25)

= 10 + 20 + 5 + 5 + 8 + 55 + 44 + 31 + 3

= **141 tracks**

SCAN SCHEDULING ALGORITHM

The total number of head movements for Scan Disk Scheduling Algorithm

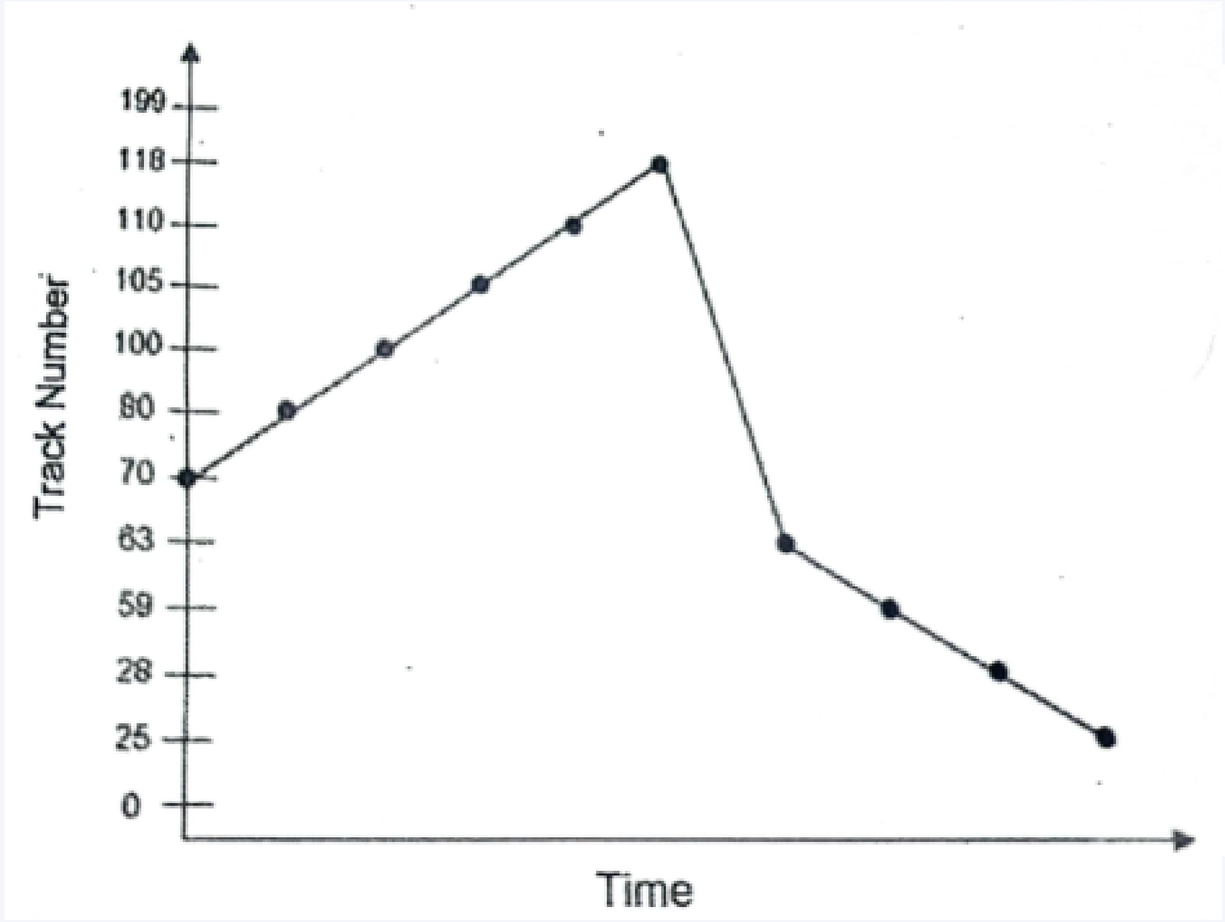
= (80 - 70) + (100 - 80) + (105 - 100) + (110 - 105) + (118 - 110) + (199 - 118) + (199 - 63) + (63 - 59) + (69 - 28) + (28 - 25)

= 10 + 20 + 5 + 5 + 8 + 81 + 136 + 4 + 31 + 3

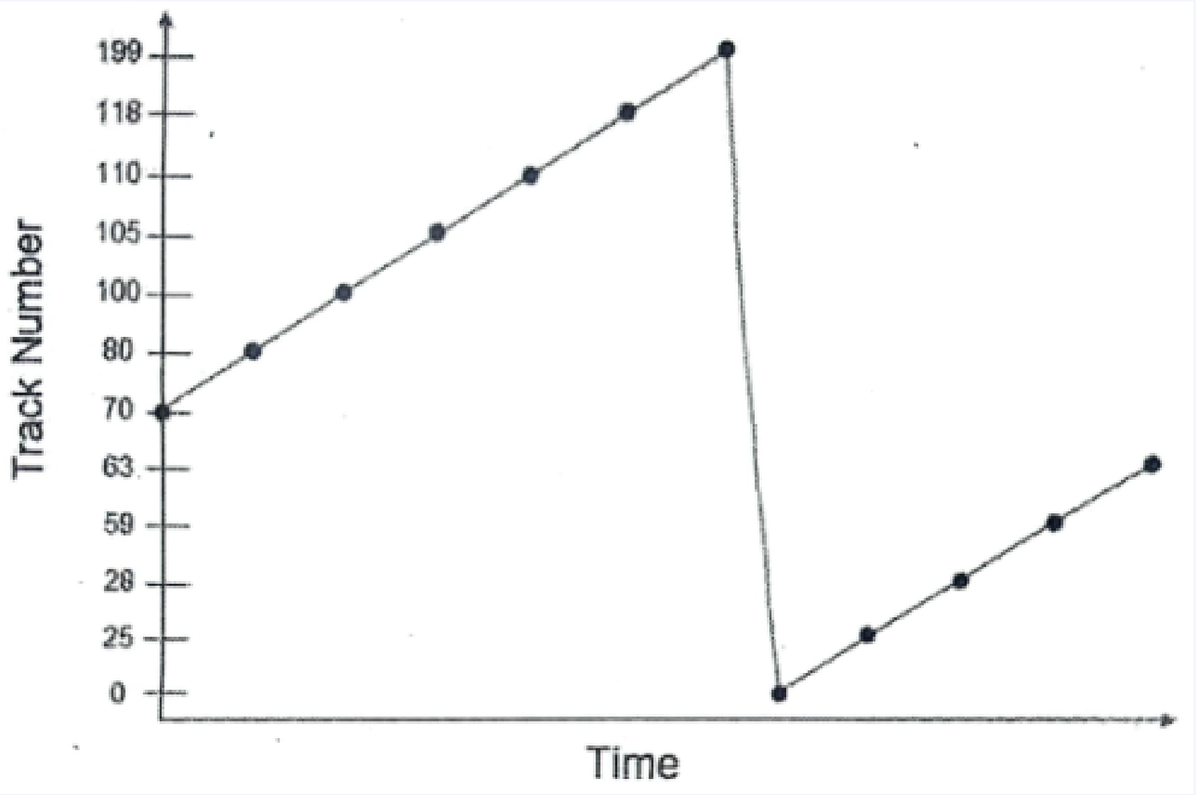
= **303 tracks**

Comparisons of Disk Scheduling Algorithms

LOOK SCHEDULING ALGORITHM



C-SCAN SCHEDULING ALGORITHM



LOOK SCHEDULING ALGORITHM

The total number of head movements for LOOK Disk Scheduling Algorithm

= (80 - 70) + (100 - 80) + (105 - 100) + (110 - 105) + (118 - 110) + (118 - 63) + (63 - 59) + (59 - 28) + (28 - 25)

= 10 + 20 + 5 + 5 + 8 + 55 + 44 + 31 + 3

= **141 tracks**

C-SCAN SCHEDULING ALGORITHM

The total number of head movements for C-Scan Disk Scheduling Algorithm

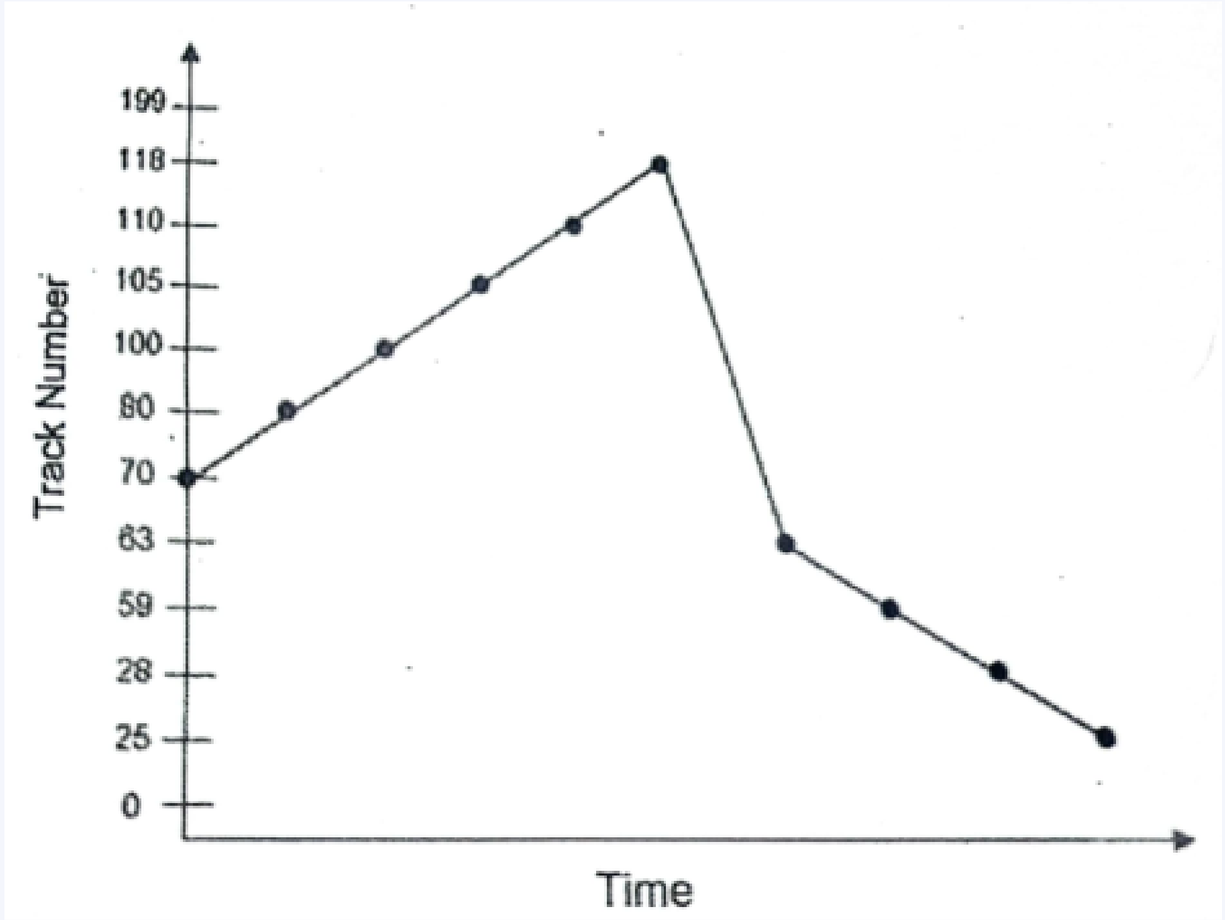
= (80 - 70) + (100 - 80) + (105 - 100) + (110 - 105) + (118 - 110) + (199 - 118) + (199 - 0) + (25 - 0) + (28 - 25) + (59 - 28) + (63 - 59)

= 10 + 20 + 5 + 5 + 8 + 81 + 199 + 25 + 3 + 31 + 4

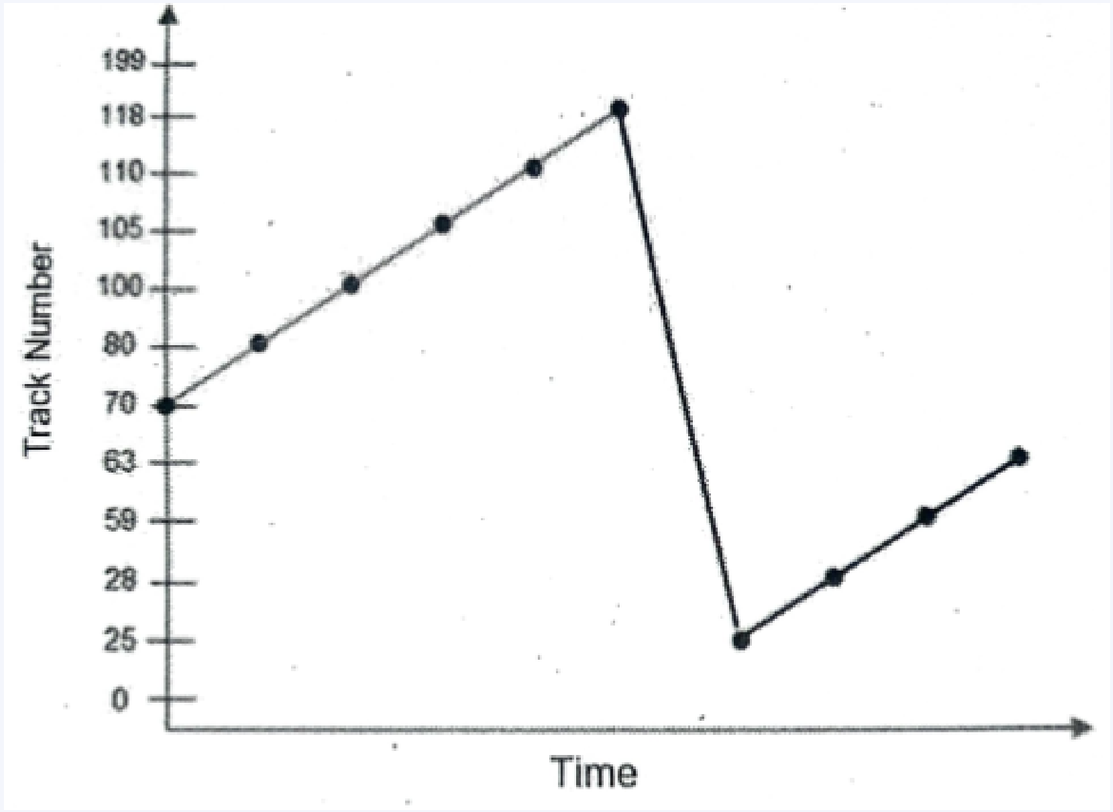
=**391 tracks**

Comparisons of Disk Scheduling Algorithms

LOOK SCHEDULING ALGORITHM



C-LOOK SCHEDULING ALGORITHM



LOOK SCHEDULING ALGORITHM

The total number of head movements for LOOK Disk Scheduling Algorithm

= (80 - 70) + (100 - 80) + (105 - 100) + (110 - 105) + (118 - 110) + (118 - 63) + (63 - 59) + (59 - 28) + (28 - 25)

= 10 + 20 + 5 + 5 + 8 + 55 + 44 + 31 + 3

= **141 tracks**

C-LOOK SCHEDULING ALGORITHM

The total number of head movements for C-LOOK Disk Scheduling Algorithm

= (80 - 70) + (100 - 80) + (105 - 100) + (110 - 105) + (118 - 110) + (118 - 25) + (28 - 25) + (59 - 28) + (63 - 28)

= 10 + 20 + 5 + 5 + 8 + 93 + 3 + 31 + 35

=**210 tracks**

Comparisons of Disk Scheduling Algorithms

Algorithm	Description	Advantages	Disadvantages
First Come, First Serve (FCFS)	Services requests in the order they arrive.	Simple to implement.	Can lead to long wait times and poor performance, especially with large seek distances.
Shortest Seek Time First (SSTF)	Services the request with the minimum seek time from the current head position.	Reduces average seek time.	Can lead to starvation, where some requests may wait indefinitely.
SCAN (Elevator) Algorithm	Scans the disk from one end to the other, servicing requests along the way. Once it reaches the end, it reverses direction.	Reduces seek time and avoids starvation.	Can lead to longer wait times for requests near the end of the scan direction.

Algorithm	Description	Advantages	Disadvantages
C-SCAN (Circular SCAN)	Scans the disk from one end to the other, servicing requests along the way. Once it reaches the end, it jumps to the opposite end without servicing any requests and then continues scanning.	Reduces starvation, improves fairness.	Can lead to longer seek times and higher overhead compared to SCAN.
LOOK	Similar to SCAN, but the head reverses direction when there are no more requests in the current direction.	Reduces unnecessary head movement compared to SCAN.	Can still lead to some variance in response time.
C-LOOK	Similar to C-SCAN, but the head reverses direction when there are no more requests in the current direction.	Improves fairness and reduces variance in response time compared to LOOK.	Can lead to longer seek times and higher overhead compared to LOOK.

REFERENCES/SOURCES

- https://www.geeksforgeeks.org/look-disk-scheduling-algorithm/
- https://www.baeldung.com/cs/disk-scheduling-look-clook
- https://workat.tech/core-cs/tutorial/disk-scheduling-algorithms-in-operating-system-os-ope5ahnn6mhh
- https://www.educative.io/answers/what-is-the-look-disk-scheduling-algorithm

THANK YOU!

- Silvestrece, Shan
- Suarez, Lyken
- Suyom, Christian
- Tamano, Hassiem