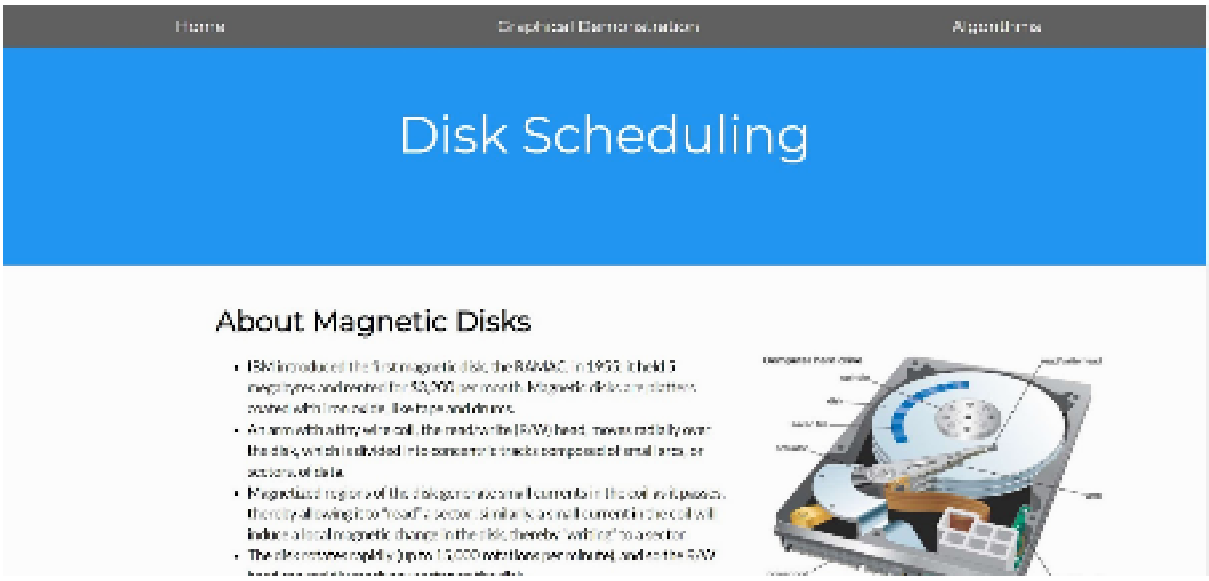
Disk Scheduling Algorithms Website

There are mainly 3 pages for the website:

1) HomePage:

The Home Page provides some basic information about magnetic disks and the need for disk scheduling.

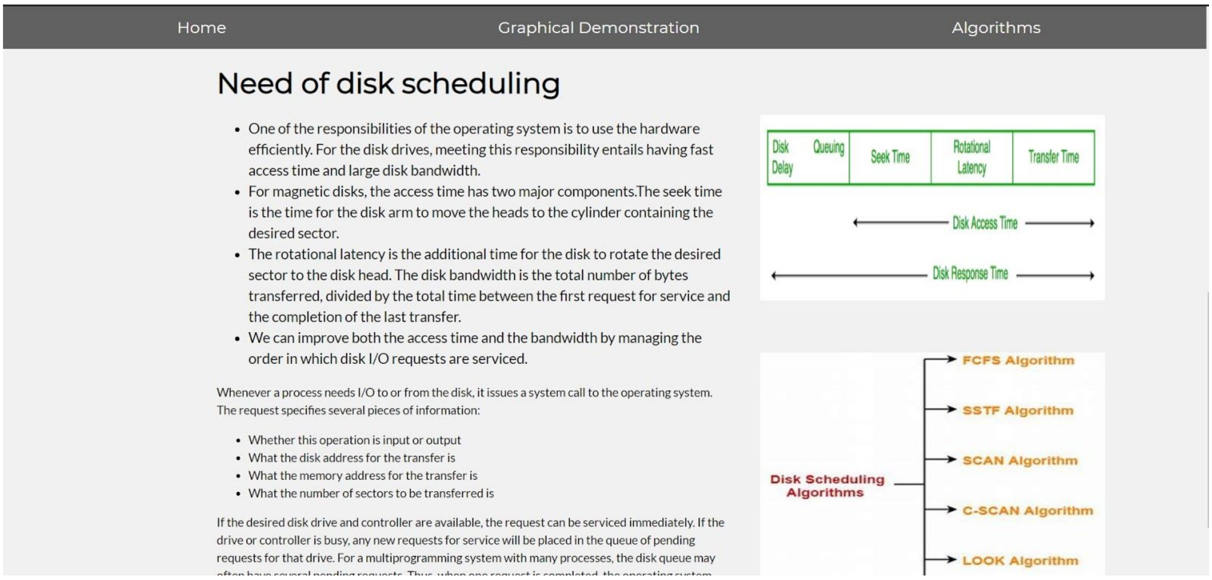
SNAPSHOTS:



Disk Scheduling

About Magnetic Disks

- IBM introduced the first magnetic disk, the RAMAC, in 1955. It held 5 megabytes and rented for $3,200 per month. Magnetic disks are platters coated with iron oxide, like tape and drums.
- An arm with a tiny wire coil, the read/write (R/W) head, moves radially over the disk, which is divided into concentric tracks composed of small arcs, or sectors, of data.
- Magnetized regions of the disk generate small currents in the coil as it passes, thereby allowing it to "read" a sector; similarly, a small current in the coil will induce a local magnetic change in the disk, thereby "writing" to a sector.
- The disk rotates rapidly (up to 15,000 rotations per minute), and so the R/W head can rapidly reach any sector on the disk.



Need of disk scheduling

- One of the responsibilities of the operating system is to use the hardware efficiently. For the disk drives, meeting this responsibility entails having fast access time and large disk bandwidth.
- For magnetic disks, the access time has two major components.The seek time is the time for the disk arm to move the heads to the cylinder containing the desired sector.
- The rotational latency is the additional time for the disk to rotate the desired sector to the disk head. The disk bandwidth is the total number of bytes transferred, divided by the total time between the first request for service and the completion of the last transfer.
- We can improve both the access time and the bandwidth by managing the order in which disk I/O requests are serviced.

Whenever a process needs I/O to or from the disk, it issues a system call to the operating system. The request specifies several pieces of information:

- Whether this operation is input or output
- What the disk address for the transfer is
- What the memory address for the transfer is
- What the number of sectors to be transferred is

If the desired disk drive and controller are available, the request can be serviced immediately. If the drive or controller is busy, any new requests for service will be placed in the queue of pending requests for that drive. For a multiprogramming system with many processes, the disk queue may often have several pending requests. Thus, when one request is completed, the operating system

2) Algorithms Page:

It contains basic information about all the disk scheduling algorithms.
The necessary info regarding all the algorithms is given below:

- FCFS Disk Scheduling Algorithms

FCFS stands for First Come First Serve. It is the simplest Disk Scheduling algorithm. As the name suggests, this algorithm entertains requests in the order they arrive in the disk queue.

Advantages-

It is simple and easy to understand. The algorithm looks very fair and there is no starvation (all requests are serviced sequentially) but generally, it does not provide the fastest service.

Disadvantages-

The scheme does not optimize the seek time. The request may come from different processes therefore there is the possibility of inappropriate movement of the head.

- SSTF Disk Scheduling Algorithm

SSTF stands for Shortest Seek Time First. This algorithm services that request next which requires the least number of head movements from its current position regardless of the direction. It breaks the tie in the direction of head movement.

Advantages-

It reduces the total seek time as compared to FCFS. It provides increased throughput. It provides less average response time and waiting time.

Disadvantages-

There is an overhead of finding out the closest request. The requests which are far from the head might starve for the CPU. It provides a high variance in response time and waiting time. Switching the direction of the head frequently slows down the algorithm.

- SCAN Disk Scheduling Algorithm

  SCAN Disk Scheduling Algorithm- As the name suggests, this algorithm scans all the cylinders of the disk back and forth. Head starts from one end of the disk and move towards the other end servicing all the requests in between. After reaching the other end, the head reverses its direction and move towards the starting end servicing all the requests in between. The same process repeats.

  NOTES-

  SCAN Algorithm is also called an Elevator Algorithm. This is because its working resembles the working of an elevator.

  Advantages-

  It is simple, easy to understand, and implement. It does not lead to starvation. It provides low variance in response time and waiting time.

  Disadvantages-

  It causes a long waiting time for the cylinders just visited by the head. It causes the head to move till the end of the disk even if there are no requests to be serviced.

- C - SCAN Disk Scheduling Algorithm

  C - SCAN Disk Scheduling Algorithm- Circular-SCAN Algorithm is an improved version of the SCAN Algorithm. Head starts from one end of the disk and move towards the other end servicing all the requests in between. After reaching the other end, the head reverses its direction. It then returns to the starting end without servicing any request in between. The same process repeats.

  Advantages-

  The waiting time for the cylinders just visited by the head is reduced as compared to the SCAN Algorithm. It provides a uniform waiting time. It provides better response time.

  Disadvantages-

It causes more seek movements as compared to SCAN Algorithm. It causes the head to move till the end of the disk even if there are no requests to be serviced.

- <u>LOOK Disk Scheduling Algorithm</u>

  LOOK Disk Scheduling Algorithm- The LOOK Algorithm is an improved version of the SCAN Algorithm. Head starts from the first request at one end of the disk and moves towards the last request at the other end servicing all the requests in between. After reaching the last request at the other end, the head reverses its direction. It then returns to the first request at the starting end servicing all the requests in between. The same process repeats.

  NOTE-

  The main difference between SCAN Algorithm and LOOK Algorithm is- SCAN Algorithm scans all the cylinders of the disk starting from one end to the other end even if there are no requests at the ends. LOOK Algorithm scans all the cylinders of the disk starting from the first request at one end to the last request at the other end.

  Advantages-

  It does not cause the head to move till the ends of the disk when there are no requests to be serviced. It provides better performance as compared to SCAN Algorithm. It does not lead to starvation. It provides low variance in response time and waiting time.

  Disadvantages-

  There is an overhead of finding the end requests. It causes a long waiting time for the cylinders just visited by the head.

- <u>C - LOOK Disk Scheduling Algorithm</u>

  C - LOOK Disk Scheduling Algorithm- Circular-LOOK Algorithm is an improved version of the LOOK Algorithm. Head starts from the first request at one end of the disk and moves towards the last request at the other end servicing all the requests in between. After reaching the last request at the other end, the head reverses its direction. It then returns to the first request at the starting end without servicing any request in between. The same process repeats.

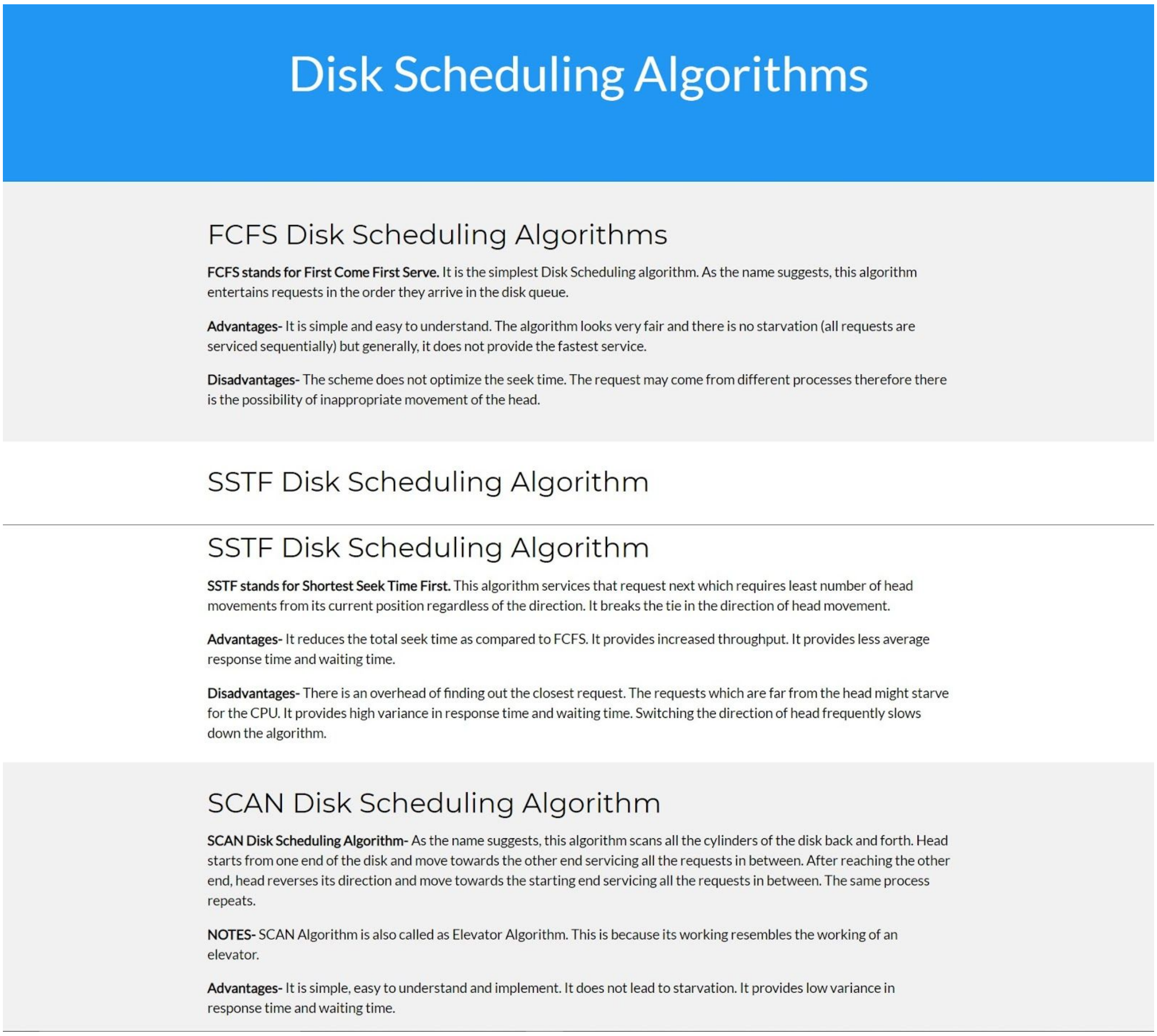
  Advantages-

It does not cause the head to move till the ends of the disk when there are no requests to be serviced. It reduces the waiting time for the cylinders just visited by the head. It provides better performance as compared to LOOK Algorithm. It does not lead to starvation. It provides low variance in response time and waiting time.

Disadvantages-

There is an overhead of finding the end requests.

SNAPSHOTS:



Disk Scheduling Algorithms

FCFS Disk Scheduling Algorithms

FCFS stands for First Come First Serve. It is the simplest Disk Scheduling algorithm. As the name suggests, this algorithm entertains requests in the order they arrive in the disk queue.

Advantages- It is simple and easy to understand. The algorithm looks very fair and there is no starvation (all requests are serviced sequentially) but generally, it does not provide the fastest service.

Disadvantages- The scheme does not optimize the seek time. The request may come from different processes therefore there is the possibility of inappropriate movement of the head.

SSTF Disk Scheduling Algorithm

SSTF Disk Scheduling Algorithm

SSTF stands for Shortest Seek Time First. This algorithm services that request next which requires least number of head movements from its current position regardless of the direction. It breaks the tie in the direction of head movement.

Advantages- It reduces the total seek time as compared to FCFS. It provides increased throughput. It provides less average response time and waiting time.

Disadvantages- There is an overhead of finding out the closest request. The requests which are far from the head might starve for the CPU. It provides high variance in response time and waiting time. Switching the direction of head frequently slows down the algorithm.

SCAN Disk Scheduling Algorithm

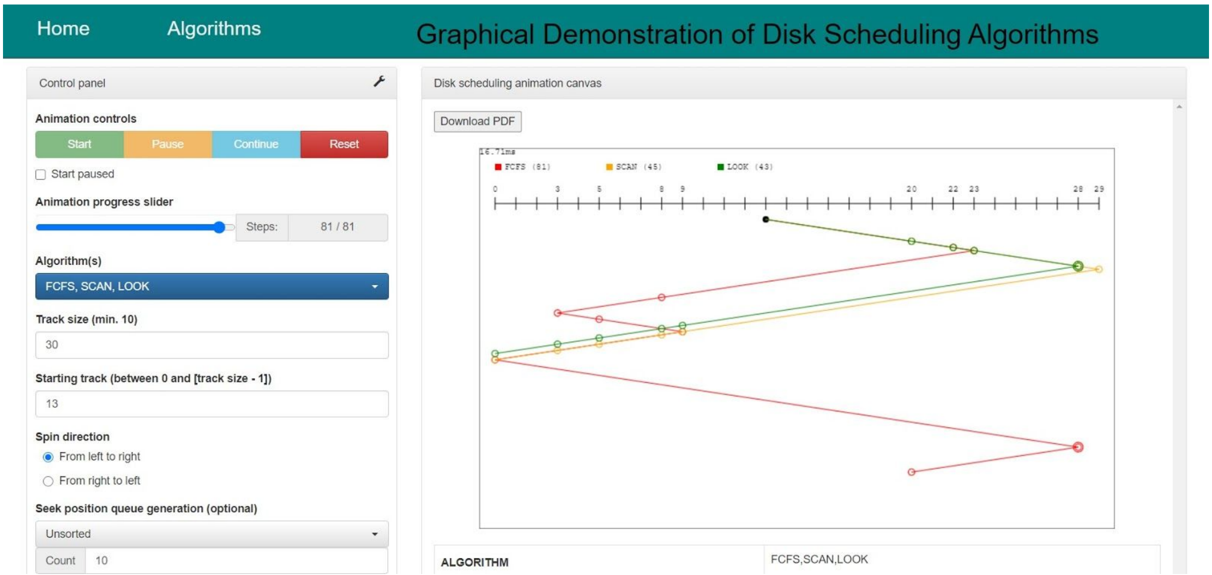
SCAN Disk Scheduling Algorithm- As the name suggests, this algorithm scans all the cylinders of the disk back and forth. Head starts from one end of the disk and move towards the other end servicing all the requests in between. After reaching the other end, head reverses its direction and move towards the starting end servicing all the requests in between. The same process repeats.

NOTES- SCAN Algorithm is also called as Elevator Algorithm. This is because its working resembles the working of an elevator.

Advantages- It is simple, easy to understand and implement. It does not lead to starvation. It provides low variance in response time and waiting time.

3) Demonstration page:

This page is the main demonstration page of all the algorithms. It also provides live visualization of each algorithm where you can provide input of your choice and get output for a specific algorithm, even more than 1 algorithm at once



☐ Start paused

Animation progress slider — Steps: 81 / 81

Algorithm(s): FCFS, SCAN, LOOK

Track size (min. 10): 30

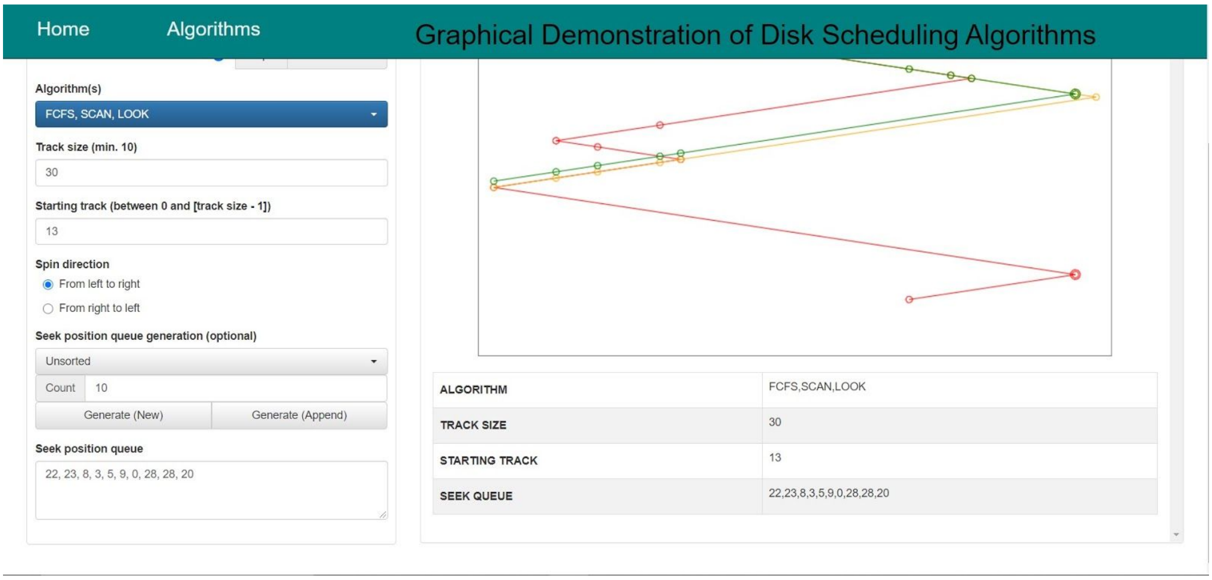
Starting track (between 0 and [track size - 1]): 13

Spin direction
● From left to right
○ From right to left

Seek position queue generation (optional): Unsorted

Count: 10

ALGORITHM	FCFS,SCAN,LOOK



Algorithm(s): FCFS, SCAN, LOOK

Track size (min. 10): 30

Starting track (between 0 and [track size - 1]): 13

Spin direction
● From left to right
○ From right to left

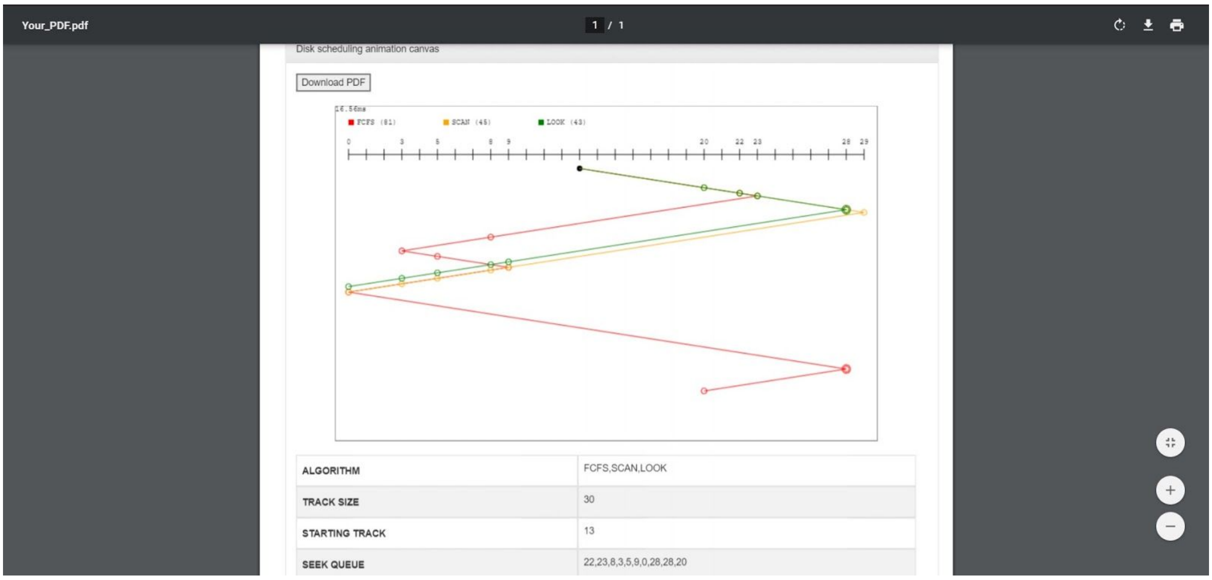
Seek position queue generation (optional): Unsorted

Count: 10

Seek position queue: 22, 23, 8, 3, 5, 9, 0, 28, 28, 20

ALGORITHM	FCFS,SCAN,LOOK
TRACK SIZE	30
STARTING TRACK	13
SEEK QUEUE	22,23,8,3,5,9,0,28,28,20

There Is also a feature of downloading your input and output of the demonstration in the form of a PDF file, which will be saved locally to your device.



ALGORITHM	FCFS,SCAN,LOOK
TRACK SIZE	30
STARTING TRACK	13
SEEK QUEUE	22,23,8,3,5,9,0,28,28,20