



PDEU PANDIT
DEENDAYAL
ENERGY
UNIVERSITY

Formerly Pandit Deendayal Petroleum University (PDPU)

DISK SCHEDULING ALGORITHM

OPERATING SYSTEM LAB

TEAM NO: 11 (READ ME FILE)

TEAM MEMBER'S:

ARYA PATEL 19IT014

BHAVYA SONI 19BIT019

DEVANSHI PATEL 19BIT026

DWARKESH GOHIL 19BIT038

JEEL JAIN 19BIT054

JEET DAVDA 19BIT055

■ OVERVIEW:

This website demonstrates various types of Disk Scheduling algorithm. User has to enter the current track number and the queue for the algorithm and as output it can be analyzed how the algorithm works with the graph. Moreover this website also includes comparison of seek time between all the six algorithm.

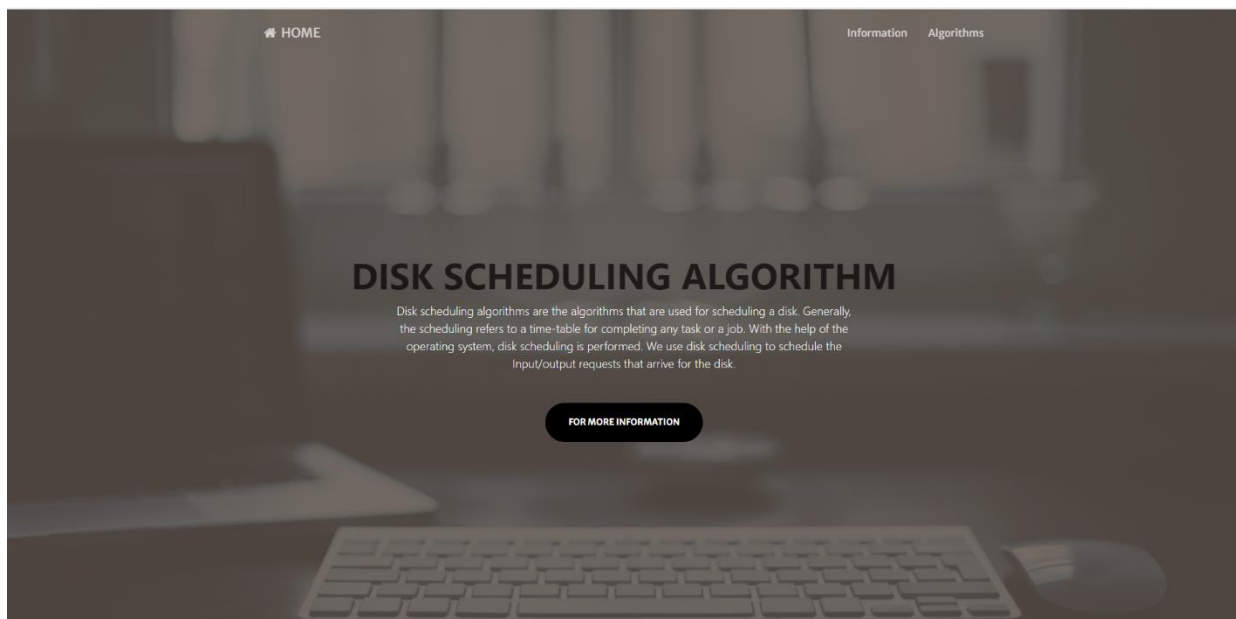
■ TECHNICAL SPECIFICATION:

LANGUAGE: HTML, CSS, JavaScript

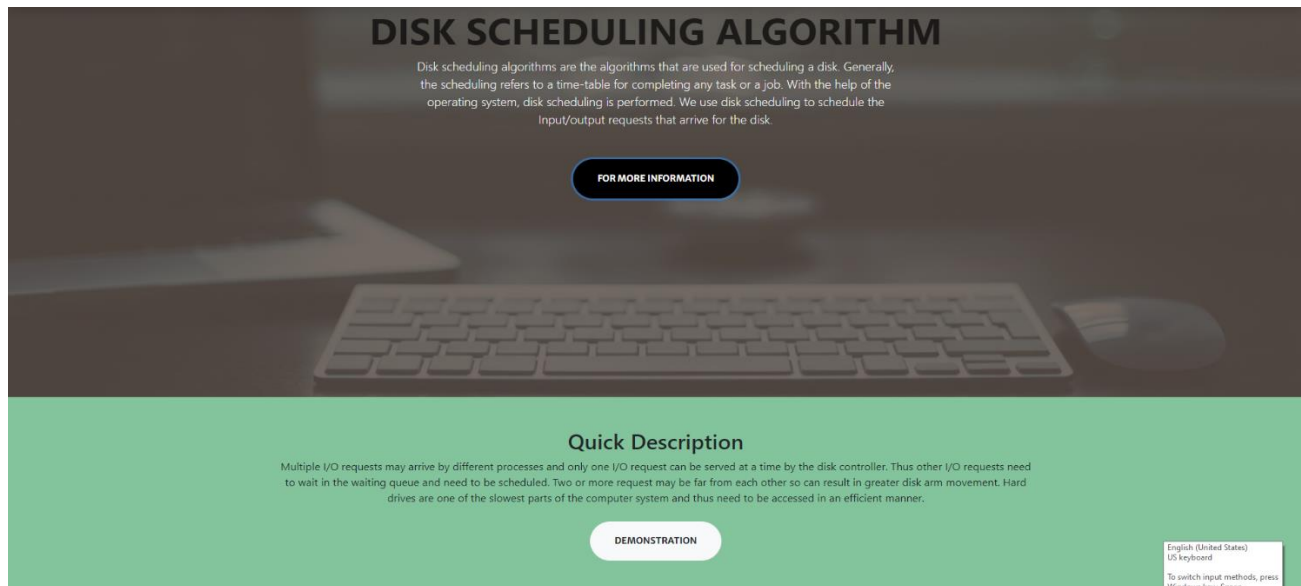
IDE: VS CODE

■ HOW TO RUN?

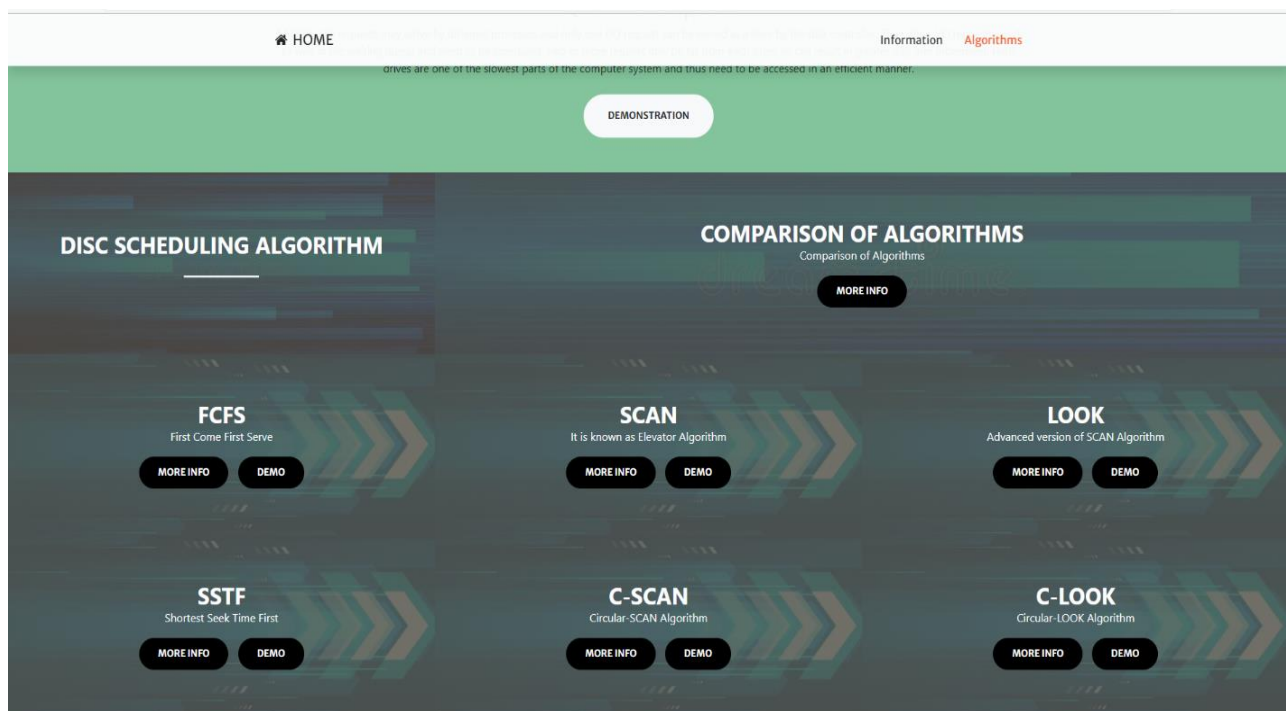
- Click on <https://jeeljain19.github.io/team11/> this link.
- You will be directed to the home page of algorithms (**DISK SCHEDULING ALGORITHM**) dealing with respective situation.



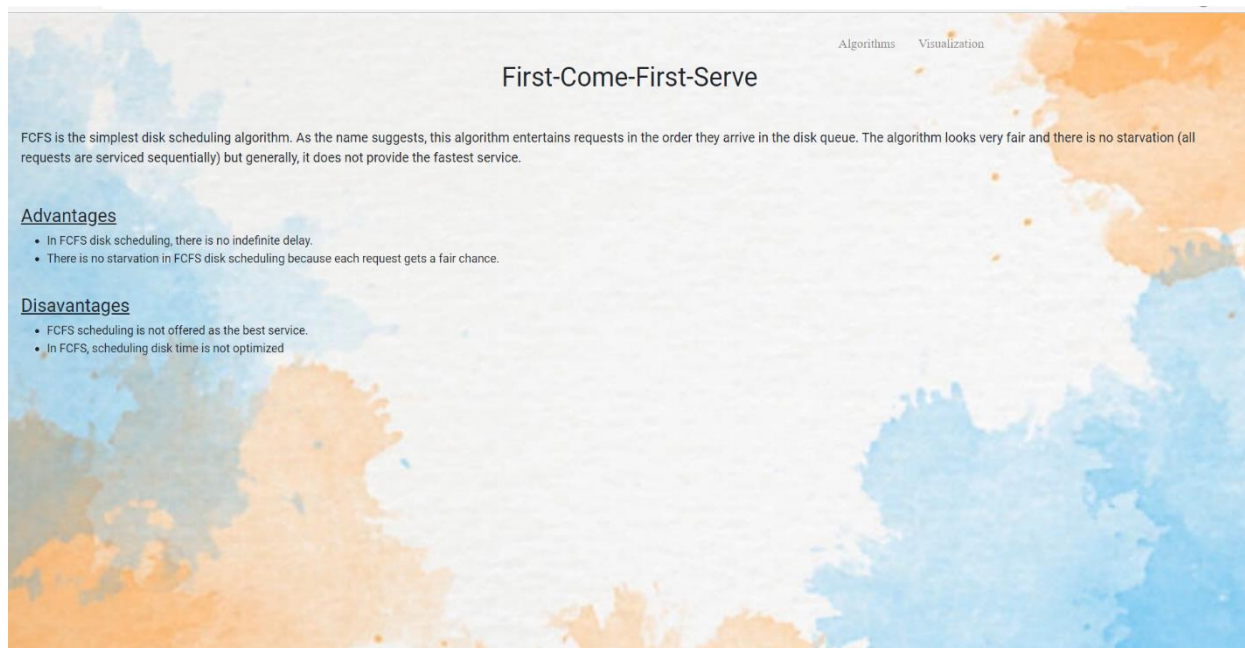
- By clicking “**Find more information**” you would be directed to the information related to Disk Scheduling algorithms



- By clicking on “**Algorithms**” you would be directed to the Algorithms (FCFS, SSTF, LOOK, SCAN, CSCAN Algorithm etc.) as shown below.



- In each Algorithm there are two options named as “**more info**” and “**demo**”.
- You can select any algorithm you want to run. Let us take an example of FCFS Algorithm. In FCFS Algorithm, as user clicks on “**more info**” it gets directed to another page .
- This page includes information about the algorithm along with its advantages and disadvantages.



- Click on “**visualization**” given in the navigation bar.
- Now, you would be directed to another page of respective algorithm. This page is designed to demonstrate the implementation of the algorithm.
- This page requires two inputs “Current Track Position” and “Track Number” from user. After entering the above the data, click on “**SUBMIT**” to send the data.
- Next, click on “**CALCULATE**” to compute the seek time of the given data for respective algorithm.

FCFS ALGORITHM

Current Track Position :

Track Number:

SUBMIT

CALCULATE

SHOW GRAPH

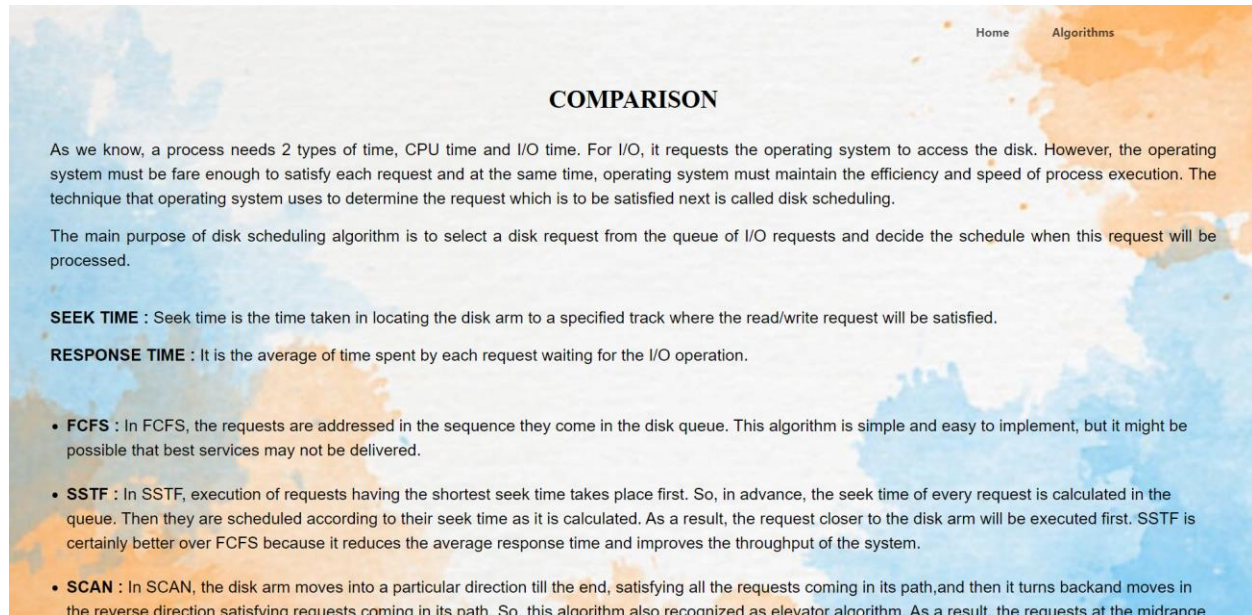
OUTPUT

- At last, clicking on “**SHOW GRAPH**” it will generate graph for the given data.



- Going back to home page, there is one more option called “**COMPARISON OF ALGORITHMS**”

- By clicking on “**MORE INFO**”, it will direct the user to new page which includes the information related to all the algorithms and their comparison.



Home Algorithms

COMPARISON

As we know, a process needs 2 types of time, CPU time and I/O time. For I/O, it requests the operating system to access the disk. However, the operating system must be fare enough to satisfy each request and at the same time, operating system must maintain the efficiency and speed of process execution. The technique that operating system uses to determine the request which is to be satisfied next is called disk scheduling.

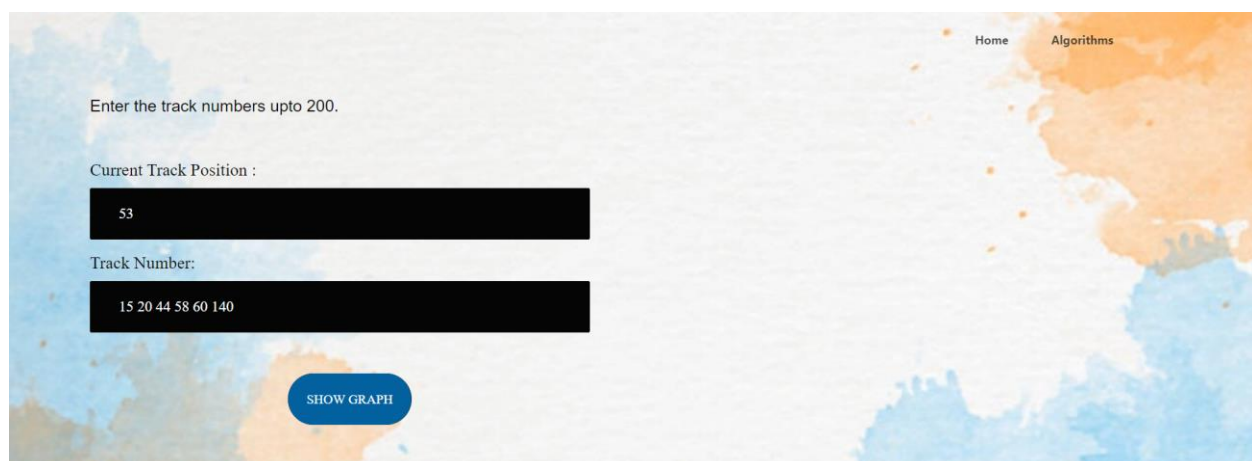
The main purpose of disk scheduling algorithm is to select a disk request from the queue of I/O requests and decide the schedule when this request will be processed.

SEEK TIME : Seek time is the time taken in locating the disk arm to a specified track where the read/write request will be satisfied.

RESPONSE TIME : It is the average of time spent by each request waiting for the I/O operation.

- **FCFS** : In FCFS, the requests are addressed in the sequence they come in the disk queue. This algorithm is simple and easy to implement, but it might be possible that best services may not be delivered.
- **SSTF** : In SSTF, execution of requests having the shortest seek time takes place first. So, in advance, the seek time of every request is calculated in the queue. Then they are scheduled according to their seek time as it is calculated. As a result, the request closer to the disk arm will be executed first. SSTF is certainly better over FCFS because it reduces the average response time and improves the throughput of the system.
- **SCAN** : In SCAN, the disk arm moves into a particular direction till the end, satisfying all the requests coming in its path, and then it turns back and moves in the reverse direction satisfying requests coming in its path. So, this algorithm also recognized as elevator algorithm. As a result, the requests at the midrange

- By clicking on “**DEMO**”, it will direct the user to new page where user can enter the data and see the graph for comparison of all the algorithm based on their seek time.



Home Algorithms

Enter the track numbers upto 200.

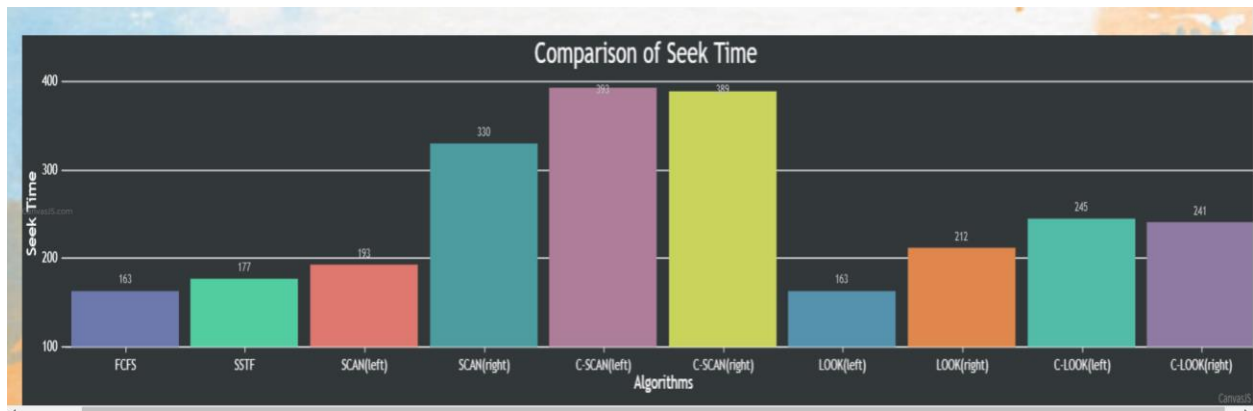
Current Track Position :

53

Track Number:

15 20 44 58 60 140

SHOW GRAPH



- In the same manner user can implement all types of disk scheduling algorithm and see the graph.