**CODE**

#include <stdio.h>

#include <stdlib.h>

// First Come First Serve

void FCFS(){

int n, i, head;

printf("\nFCFS:\nEnter the size of request array:\t");

scanf("%d", &n);

int arr[n];

printf("Enter the elements of the requested array:\t");

for(i=0; i<n; i++){

scanf("%d", &arr[i]);

}

printf("Enter the current position of disk head:\t");

scanf("%d", &head);

int seek\_count=0; int current\_track;

for(int i = 0; i < n; i++){

current\_track = arr[i];

seek\_count+=abs(current\_track-head);

head=current\_track;

}

printf("\nTotal number of seek operations is:\t%d", seek\_count);

printf("\nSeek Sequence: ");

for(i = 0; i < n; i++){

printf("%d ", arr[i]);

}

}

// Shortest Seek Time First

void difference(int req\_arr[], int diff[], int head, int n){

for(int i = 0; i < n; i++){

diff[i]=abs(head-req\_arr[i]);

}

}

int min\_diff(int req\_arr[], int visited[], int diff[], int head, int n){

int index = 0;

int min = 9999;

for(int i = 0; i < n; i++){

if(visited[i]==0 && diff[i]<min){

min=diff[i];

index=i;

}

}

return index;

}

void SSTF(){

int n, i, head, index;

printf("\nShortest Seek Time First:\nEnter the size of request array:\t");

scanf("%d", &n);

int arr[n];

printf("Enter the elements of the requested array:\t");

for(i=0; i<n; i++){

scanf("%d", &arr[i]);

}

printf("Enter the current position of disk head:\t");

scanf("%d", &head);

int visited[100], diff[100];

for(i=0; i<n; i++){

visited[i]=0;

}

int seek\_count=0; int seek\_seq[n+1];

for(i = 0; i<n; i++){

seek\_seq[i]=head;

difference(arr, diff, head, n);

index=min\_diff(arr, visited, diff, head, n);

visited[index]=1;

seek\_count+=diff[index];

head=arr[index];

}

seek\_seq[n]=head;

printf("\n\nTotal Number of Seek Operations:\t%d", seek\_count);

printf("\nSeek Sequence is:\t");

for(i=0; i<n; i++)

printf("%d ", seek\_seq[i]);

}

// Scan

void sort(int arr[], int size){

int swapped=0;

int temp;

do{

swapped=0;

for(int i=0; i<size-1; i++){

if(arr[i]>arr[i+1]){

temp = arr[i];

arr[i]=arr[i+1];

arr[i+1]=temp;

swapped=1;

}

}

}while(swapped==1);

}

void SCAN(){

int n, i, head, disk\_size, direction;

printf("\nScan:\nEnter the size of request array:\t");

scanf("%d", &n);

printf("Enter the disk size:\t");

scanf("%d", &disk\_size);

int arr[n];

printf("Enter the elements of the requested array:\t");

for(i=0; i<n; i++){

scanf("%d", &arr[i]);

}

printf("Enter the current position of disk head:\t");

scanf("%d", &head);

printf("\nChoose a direction:\n 1 for Left\n 2 for Right\nOption Selected: ");

scanf("%d", &direction);

int seek\_count = 0;

int seek\_seq[n];

int l=0, r=0, seq=0;

int current\_track;

int left[100], right[100];

if(direction==2){

right[r++]=disk\_size-1;

} else if(direction==1){

left[l++]=0;

}

for(i = 0; i<n; i++){

if(arr[i]<head){

left[l++]=arr[i];

}

if(arr[i]>head){

right[r++]=arr[i];

}

}

sort(left, l);

sort(right, r);

int loop=2;

while(loop>0){

if(direction==2){

for(i=0; i<r; i++){

current\_track=right[i];

seek\_seq[seq++]=current\_track;

seek\_count+=abs(current\_track-head);

head=current\_track;

}

direction=1;

} else if(direction==1){

for(i=l-1; i>=0; i--){

current\_track=left[i];

seek\_seq[seq++]=current\_track;

seek\_count+=abs(current\_track-head);

head=current\_track;

}

direction=2;

}

loop--;

}

printf("\n\nTotal Number of Seek Operations:\t%d", seek\_count);

printf("\nSeek Sequence is:\t");

for(i=0; i<n; i++)

printf("%d ", seek\_seq[i]);

}

// C-Look

void CLOOK(){

int n, i, head;

printf("\nC-Look:\nEnter the size of request array:\t");

scanf("%d", &n);

int arr[n];

printf("Enter the elements of the requested array:\t");

for(i=0; i<n; i++){

scanf("%d", &arr[i]);

}

printf("Enter the current position of disk head:\t");

scanf("%d", &head);

int seek\_count=0; int seek\_seq[n];

int l=0, r=0, seq=0;

int current\_track;

int left[100], right[100];

for(i=0; i<n; i++){

if(arr[i]<head){

left[l++]=arr[i];

}

if(arr[i]>head){

right[r++]=arr[i];

}

}

sort(left, l);

sort(right, r);

for(i=0; i<r; i++){

current\_track=right[i];

seek\_seq[seq++]=current\_track;

seek\_count+=abs(current\_track-head);

head=current\_track;

}

seek\_count+=abs(head-left[0]);

head=left[0];

for(i=0; i<l; i++){

current\_track=left[i];

seek\_seq[seq++]=current\_track;

seek\_count+=abs(current\_track-head);

head=current\_track;

}

printf("\n\nTotal Number of Seek Operations:\t%d", seek\_count);

printf("\nSeek Sequence is:\t");

for(i=0; i<n; i++)

printf("%d ", seek\_seq[i]);

}

int main(){

int choice = 0;

do

{

printf("\n\n---------------Disk Scheduling Algrotihms---------------\n");

printf("Press 1 for FCFS\nPress 2 for SSTF\nPress 3 for SCAN\nPress 4 for C-LOOK\nPress 5 to Exit\nChoice:\t");

scanf("%d", &choice);

switch (choice)

{

case 1:

FCFS();

break;

case 2:

SSTF();

break;

case 3:

SCAN();

break;

case 4:

CLOOK();

break;

case 5:

printf("\n\n-----------------------THANK YOU------------------------\n\n");

exit(0);

default:

printf("\n\nEnter options between 1-5.");

break;

}

} while (choice != 5);

return 0;

}

/\* **OUTPUT**

smeet@smeet-VirtualBox:~/OSL/Assignment 8$ ./a.out

---------------Disk Scheduling Algrotihms---------------

Press 1 for FCFS

Press 2 for SSTF

Press 3 for SCAN

Press 4 for C-LOOK

Press 5 to Exit

Choice:

1

FCFS:

Enter the size of request array: 8

Enter the elements of the requested array: 176 79 34 60 92 11 41 114

Enter the current position of disk head: 50

Total number of seek operations is: 510

Seek Sequence: 176 79 34 60 92 11 41 114

---------------Disk Scheduling Algrotihms---------------

Press 1 for FCFS

Press 2 for SSTF

Press 3 for SCAN

Press 4 for C-LOOK

Press 5 to Exit

Choice: 2

Shortest Seek Time First:

Enter the size of request array: 8

Enter the elements of the requested array: 176 79 34 60 92 11 41 114

Enter the current position of disk head: 50

Total Number of Seek Operations: 204

Seek Sequence is: 50 41 34 11 60 79 92 114

---------------Disk Scheduling Algrotihms---------------

Press 1 for FCFS

Press 2 for SSTF

Press 3 for SCAN

Press 4 for C-LOOK

Press 5 to Exit

Choice: 3

Scan:

Enter the size of request array: 8

Enter the disk size: 200

Enter the elements of the requested array: 176 79 34 60 92 11 41 114

Enter the current position of disk head: 50

Choose a direction:

1 for Left

2 for Right

Option Selected: 1

Total Number of Seek Operations: 226

Seek Sequence is: 41 34 11 0 60 79 92 114

---------------Disk Scheduling Algrotihms---------------

Press 1 for FCFS

Press 2 for SSTF

Press 3 for SCAN

Press 4 for C-LOOK

Press 5 to Exit

Choice: 3

Scan:

Enter the size of request array: 8

Enter the disk size: 200

Enter the elements of the requested array: 176 79 34 60 92 11 41 114

Enter the current position of disk head: 50

Choose a direction:

1 for Left

2 for Right

Option Selected: 2

Total Number of Seek Operations: 337

Seek Sequence is: 60 79 92 114 176 199 41 34

---------------Disk Scheduling Algrotihms---------------

Press 1 for FCFS

Press 2 for SSTF

Press 3 for SCAN

Press 4 for C-LOOK

Press 5 to Exit

Choice: 4

C-Look:

Enter the size of request array: 8

Enter the elements of the requested array: 176 79 34 60 92 11 41 114

Enter the current position of disk head: 50

Total Number of Seek Operations: 321

Seek Sequence is: 60 79 92 114 176 11 34 41

---------------Disk Scheduling Algrotihms---------------

Press 1 for FCFS

Press 2 for SSTF

Press 3 for SCAN

Press 4 for C-LOOK

Press 5 to Exit

Choice: 5

-----------------------THANK YOU------------------------

\*/

**OUTPUT-Screenshots**



 