// SSTF,SCAN,C_LOOC

```c
#include<stdio.h>
#include<stdlib.h>
#include<math.h>
int choice,track,no_req,head,head1,distance;
int disc_req[100],finish[100];
void menu()
{
 printf("\n\n********MENU********");
 printf("\n 1. Input data\n 2. SSTF \n 3. SCAN \n 4. C-LOOK \n 5. Exit");
 printf("\n\n Enter your choice \n");
 scanf("%d",&choice);
}
void input()
{
 int i;
 printf("Enter Total number of tracks");
 scanf("%d",&track);
 printf("Enter total number of disc requests");
 scanf("%d",&no_req);
 printf("\n Enter disc requests in FCFS order");
 for(i=0;i<no_req;i++)
 {
 scanf("%d",&disc_req[i]);

 }
 printf("\n Enter current head position");
 scanf("%d",&head1);
}
void sstf()
{
 int min,diff;
 int pending=no_req;
 int i,distance=0,index;
 head=head1;

 for(i=0;i<no_req;i++)
 {
 finish[i]=0;
 }
 printf("\n%d=>",head);
 while(pending>0)
 { min=9999;
 for(i=0;i<no_req;i++)
 {
 diff=abs(head-disc_req[i]);
```

```
 if(finish[i]==0 && diff<min)
 {
 min=diff;
 index=i;
 }
 }
 finish[index]=1;
 distance+=abs(head-disc_req[index]);
 head=disc_req[index];
 pending--;
 printf("%d=>",head);
 }
 printf("End");
 printf("\n\n Total Distance Traversed=%d",distance);
printf("\n Average Distance = %f", (float)distance/no_req);
}
void sort()
{
 int i,j,temp;
 for(i=0;i<no_req;i++)
 {
 for(j=0;j<no_req;j++)
 {
 if(disc_req[i]<disc_req[j])
 {
 temp=disc_req[i];
 disc_req[i]=disc_req[j];
 disc_req[j]=temp;
 }
 }
 }
}
void scan()
{
 int index,dir;
 int i;
 distance=0;
 head=head1;
 printf("Enter the maximum Value");
 int max;
scanf("%d",&max);
 printf("\n Enter the direction of head \n 1 - Towars higher disc(Right)\n 0 -
towards lower disc(left)");
 scanf("%d",&dir);
 sort();
 printf("\n Sorted Disc requests are: ");
 for(i=0;i<no_req;i++)
 {
```

```
printf(" %d",disc_req[i]);
}

i=0;
while(head>=disc_req[i])
{
index=i;
i++;
}
printf("\n index=%d",index);
printf("\n%d=>",head);
if(dir==1)
{
sort();
for(i=index+1;i<no_req;i++)
{
printf("%d=>",disc_req[i]);
distance+=abs(head-disc_req[i]);
head=disc_req[i];
}
distance+=abs(head-(track-1));
printf("%d=>",track-1);
head=track-1;
for(i=index;i>=0;i--)
{
printf("%d=>",disc_req[i]);
distance+=abs(head-disc_req[i]);
head=disc_req[i];
}
}
else
{
sort();
for(i=index;i>=0;i--)
{
printf("%d=>",disc_req[i]);
distance+=abs(head-disc_req[i]);
head=disc_req[i];
}
distance+=abs(head-0);
head=0;
printf("0=>");
for(i=index+1;i<no_req;i++)
{
printf("%d=>",disc_req[i]);
distance+=abs(head-disc_req[i]);
head=disc_req[i];
```

```
 }

 }
 printf("End");
 printf("\n Total Distance Traversed=%d",distance);
printf("\n Average Distance = %f", (float)distance/no_req);
}
void clook()
{
 int index,dir;
 int i;
 distance=0;
 head=head1;
 printf("\n Enter the direction of head \n 1 - Towars higher disc \n 0-towards
lower disc");
 scanf("%d",&dir);
 sort();
 printf("\n Sorted Disc requests are: ");
 for(i=0;i<no_req;i++)
 {

 printf(" %d",disc_req[i]);
 }

 i=0;
 while(head>=disc_req[i])
 {
 index=i;
 i++;
 }
 printf("\n index=%d",index);
 printf("\n%d=>",head);
 if(dir==1)
 {
 sort();
 for(i=index+1;i<no_req;i++)
 {
 printf("%d=>",disc_req[i]);
 distance+=abs(head-disc_req[i]);
 head=disc_req[i];
 }
 for(i=0;i<index;i++)
 {
 printf("%d=>",disc_req[i]);
 distance+=abs(head-disc_req[i]);
 head=disc_req[i];
 }
 }
```

```
 else
 {
 sort();
 for(i=index;i>=0;i--)
 {
 printf("%d=>",disc_req[i]);
 distance+=abs(head-disc_req[i]);
 head=disc_req[i];
 }
 for(i=(no_req-1);i>index;i--)
 {
 printf("%d=>",disc_req[i]);
 distance+=abs(head-disc_req[i]);
 head=disc_req[i];
 }

 }
 printf("End");
 printf("\n Total Distance Traversed=%d",distance);
printf("\n Average Distance = %f", (float)distance/no_req);
}
int main()
{
 while(1)
 {
 menu();
 switch(choice)
 {
 case 1: input();
 break;
 case 2: sstf();
 break;
 case 3: scan();
 break;
 case 4: clook();
 break;
 case 5: exit(0);
 break;
 default:
 printf("\n Enter valid choice");
 break;
 }
 }
 return 0;
}
```