**Pratical 08**

#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("\n1. Input data\n 2. SSTF \n 3. SCAN \n 4. C-LOOK \n 5. Exit"); printf("\n\n Enter your choice"); 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);

}

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("\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);

} 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);

}

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; }

Output :-











