***Disk Schedulling Algorithms***

***1)First Come First Serve***

///Disk Schedulling First In First Out Algorithm

///Adnan Ismail Shah Muzavor

#include<stdio.h>

#include<stdlib.h>

int i;

void FIFO(int n,int ch,int \*r,int \*e)

{

int HM=0; /// => Head Movements

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

{

if(i==0) HM+=abs(e[i]-ch);

else HM+=abs(e[i]-e[i-1]);

}

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

printf("\nBy Disk Schedulling First Come First Serve Algorithm");

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

printf("\nNumber of Disk Head Movements : %d",HM);

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

}

int main()

{

int n,Current\_head;

printf("Enter the number of elements in disk queue: ");

scanf("%d",&n);

int \*ele=(int\*)malloc(sizeof(int)\*n);

printf("\nEnter the elements in disk queue: ");

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

{

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

}

int \*range=(int\*)malloc(sizeof(int)\*2);

printf("\nEnter the range of cylinder: ");

scanf("%d",&range[0]);

scanf("%d",&range[1]);

printf("\nEnter the position of current head: ");

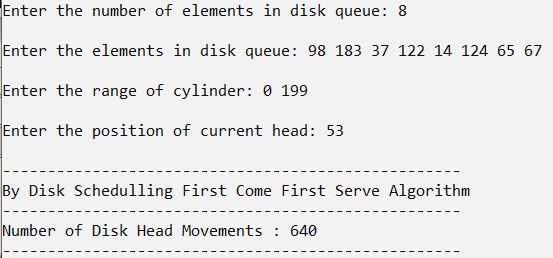
scanf("%d",&Current\_head);

FIFO(n,Current\_head,range,ele);

return 1;

}

***Output:***

******

***2)Shortest Seek Time First***

///Disk Schedulling Shortest Seek Time First

///Adnan Ismail Shah Muzavor

#include<stdio.h>

#include<stdlib.h>

#define MAX 1000;

int i;

int Give\_nearest(int \*vis,int \*ele,int n,int from\_ele)

{

int index=-1,i;

int min\_val=MAX;

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

{

if(!vis[i])

{

if(abs(from\_ele-ele[i])<min\_val)

{

min\_val=abs(from\_ele-ele[i]);

index=i;

}

}

}

return index;

}

void SSTF(int n,int ch,int \*r,int \*e)

{

int HM=0; /// => Head Movements

int \*vis=(int\*)malloc(sizeof(int)\*n);

for(i=0; i<n; i++) vis[i]=0;

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

{

//printf("THe prev ele is: %d",ch);

int index=Give\_nearest(vis,e,n,ch);

if(index==-1)

{

printf("\nUnwanted error!!");

break;

}

vis[index]=1;

HM+=abs(e[index]-ch);

ch=e[index];

}

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

printf("\nBy Disk Schedulling Shortest Seek Time First");

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

printf("\nNumber of Disk Head Movements : %d",HM);

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

}

int main()

{

int n,Current\_head;

printf("Enter the number of elements in disk queue: ");

scanf("%d",&n);

int \*ele=(int\*)malloc(sizeof(int)\*n);

printf("\nEnter the elements in disk queue: ");

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

{

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

}

int \*range=(int\*)malloc(sizeof(int)\*2);

printf("\nEnter the range of cylinder: ");

scanf("%d",&range[0]);

scanf("%d",&range[1]);

printf("\nEnter the position of current head: ");

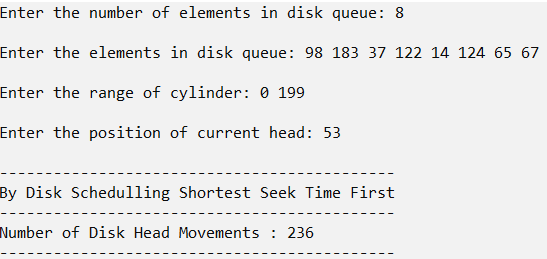
scanf("%d",&Current\_head);

SSTF(n,Current\_head,range,ele);

return 1;

}

***Output:***

******

***3)SCAN / ELEVATOR***

///Disk Schedulling SCAN / ELEVATOR

///Adnan Ismail Shah Muzavor

#include<stdio.h>

#include<stdlib.h>

#define MAX 1000;

int i;

int Min\_Val(int n,int \*ele)

{

int i,min\_ele=MAX;

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

{

if(ele[i]<min\_ele)

{

min\_ele=ele[i];

}

}

return min\_ele;

}

int Max\_Val(int n,int \*ele)

{

int i,max\_ele=-1;

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

{

if(ele[i]>max\_ele)

{

max\_ele=ele[i];

}

}

return max\_ele;

}

void SCAN(int n,int ch,int d,int \*range,int \*e)

{

int HM=0; /// => Head Movements

int \*vis=(int\*)malloc(sizeof(int)\*n);

for(i=0; i<n; i++) vis[i]=0;

if(d==1)

{

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

printf("\nDisk Direction is Towards Left");

HM=abs(range[0]-ch);

HM+=abs(range[0]-Max\_Val(n,e));

}

else

{

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

printf("\nDisk Direction is Towards Right");

HM=abs(range[1]-ch);

HM+=abs(range[1]-Min\_Val(n,e));

}

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

printf("\nBy Disk Schedulling SCAN / ELEVATOR");

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

printf("\nNumber of Disk Head Movements : %d",HM);

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

}

int main()

{

int n,Current\_head,d;

printf("Enter the number of elements in disk queue: ");

scanf("%d",&n);

int \*ele=(int\*)malloc(sizeof(int)\*n);

printf("\nEnter the elements in disk queue: ");

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

{

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

}

int \*range=(int\*)malloc(sizeof(int)\*2);

printf("\nEnter the range of cylinder: ");

scanf("%d",&range[0]);

scanf("%d",&range[1]);

printf("\nEnter the position of current head: ");

scanf("%d",&Current\_head);

printf("\nEnter the direction of disk movement (1)=>Left, (2)=>Right: ");

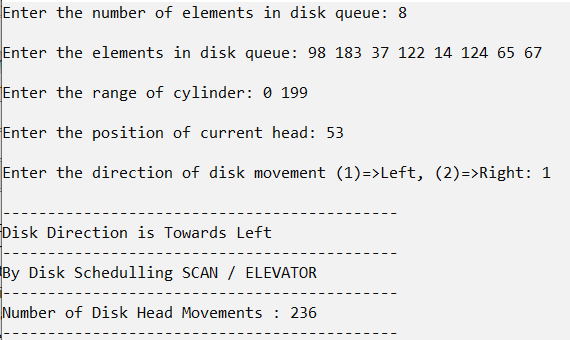
scanf("%d",&d);

SCAN(n,Current\_head,d,range,ele);

return 1;

}

***Output:***

******

***4)C-SCAN***

///Disk Schedulling C-SCAN

///Adnan Ismail Shah Muzavor

#include<stdio.h>

#include<stdlib.h>

#define MAX 1000;

int i;

int Next\_Right\_Val\_From\_CH(int n,int ch,int \*ele)

{

int val=-1,min\_diff=MAX;

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

{

if(ele[i]>=ch && abs(ele[i]-ch)<min\_diff)

{

val=ele[i];

min\_diff=abs(ele[i]-ch);

}

}

return val;

}

int Prev\_Left\_Val\_From\_CH(int n,int ch,int \*ele)

{

int val=-1,min\_diff=MAX;

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

{

if(ele[i]<=ch && abs(ele[i]-ch)<min\_diff)

{

val=ele[i];

min\_diff=abs(ele[i]-ch);

}

}

return val;

}

void SCAN(int n,int ch,int d,int \*range,int \*e)

{

int HM=0; /// => Head Movements

int \*vis=(int\*)malloc(sizeof(int)\*n);

for(i=0; i<n; i++) vis[i]=0;

if(d==1)

{

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

printf("\nDisk Direction is Towards Left");

HM=abs(range[0]-ch);

HM+=abs(range[0]-range[1]);

HM+=abs(range[1]-Next\_Right\_Val\_From\_CH(n,ch,e));

}

else

{

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

printf("\nDisk Direction is Towards Right");

HM=abs(range[1]-ch);

HM+=abs(range[0]-range[1]);

HM+=abs(range[0]-Prev\_Left\_Val\_From\_CH(n,ch,e));

}

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

printf("\nBy Disk Schedulling C-Scan Algorithm");

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

printf("\nNumber of Disk Head Movements : %d",HM);

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

}

int main()

{

int n,Current\_head,d;

printf("Enter the number of elements in disk queue: ");

scanf("%d",&n);

int \*ele=(int\*)malloc(sizeof(int)\*n);

printf("\nEnter the elements in disk queue: ");

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

{

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

}

int \*range=(int\*)malloc(sizeof(int)\*2);

printf("\nEnter the range of cylinder: ");

scanf("%d",&range[0]);

scanf("%d",&range[1]);

printf("\nEnter the position of current head: ");

scanf("%d",&Current\_head);

printf("\nEnter the direction of disk movement (1)=>Left, (2)=>Right: ");

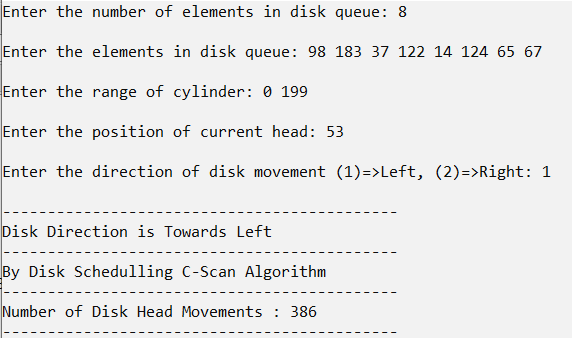
scanf("%d",&d);

SCAN(n,Current\_head,d,range,ele);

return 1;

}

***Output:***

******

***5)LOOK***

///Disk Schedulling Look Algorithm

///Adnan Ismail Shah Muzavor

#include<stdio.h>

#include<stdlib.h>

#define MAX 1000;

int i;

int Min\_Val(int n,int \*ele)

{

int i,min\_ele=MAX;

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

{

if(ele[i]<min\_ele)

{

min\_ele=ele[i];

}

}

return min\_ele;

}

int Max\_Val(int n,int \*ele)

{

int i,max\_ele=-1;

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

{

if(ele[i]>max\_ele)

{

max\_ele=ele[i];

}

}

return max\_ele;

}

void SCAN(int n,int ch,int d,int \*range,int \*e)

{

int HM=0; /// => Head Movements

int \*vis=(int\*)malloc(sizeof(int)\*n);

for(i=0; i<n; i++) vis[i]=0;

if(d==1)

{

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

printf("\nDisk Direction is Towards Left");

HM=abs(Min\_Val(n,e)-ch);

HM+=abs(Min\_Val(n,e)-Max\_Val(n,e));

}

else

{

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

printf("\nDisk Direction is Towards Right");

HM=abs(Max\_Val(n,e)-ch);

HM+=abs(Max\_Val(n,e)-Min\_Val(n,e));

}

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

printf("\nBy Disk Schedulling Look Algorithm");

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

printf("\nNumber of Disk Head Movements : %d",HM);

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

}

int main()

{

int n,Current\_head,d;

printf("Enter the number of elements in disk queue: ");

scanf("%d",&n);

int \*ele=(int\*)malloc(sizeof(int)\*n);

printf("\nEnter the elements in disk queue: ");

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

{

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

}

int \*range=(int\*)malloc(sizeof(int)\*2);

printf("\nEnter the range of cylinder: ");

scanf("%d",&range[0]);

scanf("%d",&range[1]);

printf("\nEnter the position of current head: ");

scanf("%d",&Current\_head);

printf("\nEnter the direction of disk movement (1)=>Left, (2)=>Right: ");

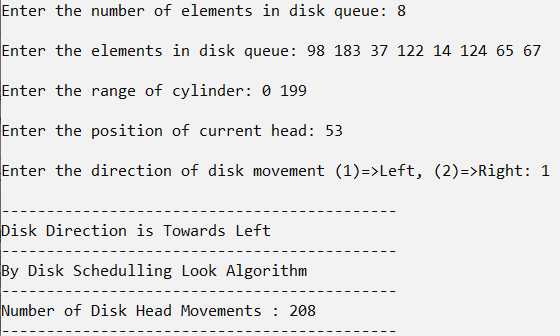
scanf("%d",&d);

SCAN(n,Current\_head,d,range,ele);

return 1;

}

***Output:***

******

***6)C-LOOK***

///Disk Schedulling C-Look

///Adnan Ismail Shah Muzavor

#include<stdio.h>

#include<stdlib.h>

#define MAX 1000;

int i;

int Min\_Val(int n,int \*ele)

{

int i,min\_ele=MAX;

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

{

if(ele[i]<min\_ele)

{

min\_ele=ele[i];

}

}

return min\_ele;

}

int Max\_Val(int n,int \*ele)

{

int i,max\_ele=-1;

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

{

if(ele[i]>max\_ele)

{

max\_ele=ele[i];

}

}

return max\_ele;

}

int Prev\_Left\_Val\_From\_CH(int n,int ch,int \*ele)

{

int val=-1,min\_diff=MAX;

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

{

if(ele[i]<=ch && abs(ele[i]-ch)<min\_diff)

{

val=ele[i];

min\_diff=abs(ele[i]-ch);

}

}

return val;

}

int Next\_Right\_Val\_From\_CH(int n,int ch,int \*ele)

{

int val=-1,min\_diff=MAX;

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

{

if(ele[i]>=ch && abs(ele[i]-ch)<min\_diff)

{

val=ele[i];

min\_diff=abs(ele[i]-ch);

}

}

return val;

}

void SCAN(int n,int ch,int d,int \*range,int \*e)

{

int HM=0; /// => Head Movements

int \*vis=(int\*)malloc(sizeof(int)\*n);

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

vis[i]=0;

if(d==1)

{

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

printf("\nDisk Direction is Towards Left");

HM=abs(Min\_Val(n,e)-ch);

HM+=abs(Min\_Val(n,e)-Max\_Val(n,e));

HM+=abs(Max\_Val(n,e)-Next\_Right\_Val\_From\_CH(n,ch,e));

}

else

{

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

printf("\nDisk Direction is Towards Right");

HM=abs(Max\_Val(n,e)-ch);

HM+=abs(Max\_Val(n,e)-Min\_Val(n,e));

HM+=abs(Min\_Val(n,e)-Prev\_Left\_Val\_From\_CH(n,ch,e));

}

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

printf("\nBy Disk Schedulling C-Look Algorithm");

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

printf("\nNumber of Disk Head Movements : %d",HM);

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

}

int main()

{

int n,Current\_head,d;

printf("Enter the number of elements in disk queue: ");

scanf("%d",&n);

int \*ele=(int\*)malloc(sizeof(int)\*n);

printf("\nEnter the elements in disk queue: ");

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

{

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

}

int \*range=(int\*)malloc(sizeof(int)\*2);

printf("\nEnter the range of cylinder: ");

scanf("%d",&range[0]);

scanf("%d",&range[1]);

printf("\nEnter the position of current head: ");

scanf("%d",&Current\_head);

printf("\nEnter the direction of disk movement (1)=>Left, (2)=>Right: ");

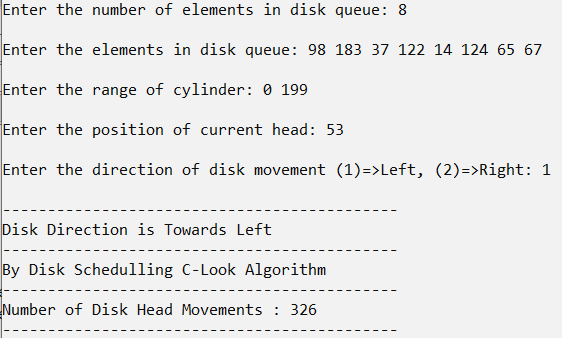
scanf("%d",&d);

SCAN(n,Current\_head,d,range,ele);

return 1;

}

***Output:***

******