assignment 2

FCFS :

```
#include<stdio.h>
#include<stdlib.h>

void fcfs(int dB[], int head, int n){
    int TotalHeadMov = 0;

    printf("\nSEEK SEQUENCE : \n");
    printf("%d\t",head);

    for(int i = 0; i < n; i++){
        printf("%d\t",dB[i]);
        TotalHeadMov += abs(dB[i] - head);
        head = dB[i];
    }

    printf("\n\nTOTAL HEAD MOVEMENTS : %d\n", TotalHeadMov);
}

int main()
{
    int dB[100],i,n,head;
    printf("Enter the number of disk blocks : \n");
    scanf("%d",&n);

    printf("Enter the Request String : \n");
    for(i = 0; i < n; i++)
     scanf("%d",&dB[i]);


    printf("Enter initial head position : \n");
    scanf("%d",&head);

    fcfs(dB, head, n);
    return 0;

}
```

SSTF :

```
#include<stdio.h>
#include<stdlib.h>
#define INT_MAX 2147483647


void sstf(int dB[], int head, int n){
    int count = 0, diff, ind, min;
```

```
    int TotalHeadMov = 0;

    printf("\nSEEK SEQUENCE : \n");
    printf("%d\t",head);

    while(count < n){
        int min = INT_MAX;

        for(int i = 0; i < n; i++){
            diff = abs(dB[i] - head);

            if(diff < min){
                min = diff;
                ind = i;
            }
        }

        printf("%d\t",dB[ind]);

        TotalHeadMov += min;

        head = dB[ind];
        dB[ind] = INT_MAX;
        count++;
    }
    printf("\n\nTOTAL HEAD MOVEMENTS : %d\n", TotalHeadMov);
}

int main()
{
    int dB[100],i,n,head;
    printf("Enter the number of disk blocks : \n");
    scanf("%d",&n);

    printf("Enter the Request String : \n");
    for(i = 0; i < n; i++)
     scanf("%d",&dB[i]);


    printf("Enter initial head position : \n");
    scanf("%d",&head);

    sstf(dB, head, n);
    return 0;

}
```

SCAN :

```
#include<stdio.h>
#include<stdlib.h>
```

```
void sort(int dB[], int n){
        for(int i = 0; i < n; i++)
    {
        for(int j = 0; j < n-i-1; j++)
        {
            if(dB[j] > dB[j+1])
            {
                int temp = dB[j];
                dB[j] = dB[j+1];
                dB[j+1] = temp;
            }
        }
    }

}

void scan(int dB[], int head, int n, int size, int direct){
    int TotalHeadMov = 0;
        sort(dB, n);

        int ind;
        for(int i = 0; i < n; i++){
                if(head < dB[i]){
                 ind = i;
                 break;
                }
        }

        printf("\nSEEK SEQUENCE : \n");
    printf("%d\t",head);

        //higher values
        if(direct == 1){
                for(int i = ind; i < n; i++){
                        printf("%d\t",dB[i]);
                        TotalHeadMov += abs(dB[i] - head);
                        head = dB[i];
                }

                //outermost track
                TotalHeadMov += abs(size - 1 - dB[n-1]);
                printf("%d\t",size-1);

                //returning
                head = size - 1;
                for(int i = ind - 1; i >= 0; i--){
                        printf("%d\t",dB[i]);
                        TotalHeadMov += abs(dB[i] - head);
                        head = dB[i];
                }
        }
        else{//lower values
                for(int i = ind - 1; i >= 0; i--){
```

```
                        printf("%d\t",dB[i]);
                        TotalHeadMov += abs(dB[i] - head);
                        head = dB[i];
                }

                //innermost track
                TotalHeadMov += dB[0]; //db[0]-0
                printf("0\t");

                //returning
                head = 0;
                for(int i = ind; i < n; i++){
                        printf("%d\t",dB[i]);
                        TotalHeadMov += abs(dB[i] - head);
                        head = dB[i];
                }
        }
    printf("\n\nTOTAL HEAD MOVEMENTS : %d\n", TotalHeadMov);
}

int main()
{
    int dB[100],i,n,head,size,direct;
    printf("Enter the number of disk blocks : \n");
    scanf("%d",&n);

    printf("Enter the Request String : \n");
    for(i = 0; i < n; i++)
     scanf("%d",&dB[i]);


    printf("Enter initial head position : \n");
    scanf("%d",&head);

    printf("Enter disk size, i.e. total number of tracks : ");
    scanf("%d",&size);

    printf("Enter direction of head movement\n1 - Higher values, 0 - Lower
values : ");
    scanf("%d",&direct);

    scan(dB, head, n, size, direct);
    return 0;

}
```

CSCAN :

```
#include<stdio.h>
#include<stdlib.h>
```

```
void sort(int dB[], int n){
        for(int i = 0; i < n; i++)
    {
        for(int j = 0; j < n-i-1; j++)
        {
            if(dB[j] > dB[j+1])
            {
                int temp = dB[j];
                dB[j] = dB[j+1];
                dB[j+1] = temp;
            }
        }
    }

}

void scan(int dB[], int head, int n, int size, int direct){
    int TotalHeadMov = 0;
        sort(dB, n);

        int ind;
        for(int i = 0; i < n; i++){
                if(head < dB[i]){
                 ind = i;
                 break;
                }
        }

        printf("\nSEEK SEQUENCE : \n");
    printf("%d\t",head);

        //higher values
        if(direct == 1){
                for(int i = ind; i < n; i++){
                        printf("%d\t",dB[i]);
                        TotalHeadMov += abs(dB[i] - head);
                        head = dB[i];
                }

                //outermost track
                TotalHeadMov += abs(size - 1 - dB[n-1]);
                printf("%d\t",size-1);

                //returning
                TotalHeadMov += size - 1; //(size -1) - 0
                printf("0\t");
                head = 0;
                for(int i = 0; i < ind; i++){
                        printf("%d\t",dB[i]);
                        TotalHeadMov += abs(dB[i] - head);
                        head = dB[i];
                }
        }
        else{//lower values
```

```
                for(int i = ind - 1; i >= 0; i--){
                        printf("%d\t",dB[i]);
                        TotalHeadMov += abs(dB[i] - head);
                        head = dB[i];
                }

                //innermost track
                TotalHeadMov += dB[0]; //db[0]-0
                printf("0\t");

                //returning
                TotalHeadMov += size - 1;
                printf("%d\t",size-1);
                head = size - 1;
                for(int i = n - 1; i >= ind; i--){
                        printf("%d\t",dB[i]);
                        TotalHeadMov += abs(dB[i] - head);
                        head = dB[i];
                }
        }
    printf("\n\nTOTAL HEAD MOVEMENTS : %d\n", TotalHeadMov);
}

int main()
{
    int dB[100],i,n,head,size,direct;
    printf("Enter the number of disk blocks : \n");
    scanf("%d",&n);

    printf("Enter the Request String : \n");
    for(i = 0; i < n; i++)
     scanf("%d",&dB[i]);


    printf("Enter initial head position : \n");
    scanf("%d",&head);

    printf("Enter disk size, i.e. total number of tracks : ");
    scanf("%d",&size);

    printf("Enter direction of head movement\n1 - Higher values, 0 - Lower
values : ");
    scanf("%d",&direct);

    scan(dB, head, n, size, direct);
    return 0;

}
```
