/\*

# PROGRAM TO IMPLEMENT DISK SCHEDULING ALGORITHMS – C-SCAN

Ashis Solomon

CS4B 17

MDL20CS035

\*/

**CODE:**

#include <stdio.h>

#include <stdlib.h>

#define MAX 25

int n, head, size, seek\_count, tracks[MAX], sequence[MAX];

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

int temp;

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

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

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

temp = arr[j];

arr[j] = arr[j + 1];

arr[j + 1] = temp;

}

}

}

}

void cscands() {

int curr\_track, distance, l, r, left[MAX], right[MAX];

seek\_count = 0;

l = 0;

r = 0;

left[0] = 0;

l++;

right[0] = size - 1;

r++;

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

if (tracks[i] < head)

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

if (tracks[i] > head)

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

}

sort(left, l);

sort(right, r);

int x = 0;

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

curr\_track = right[i];

sequence[x++] = curr\_track;

distance = abs(head - curr\_track);

seek\_count += distance;

head = curr\_track;

}

head = 0;

seek\_count += size - 1;

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

curr\_track = left[i];

sequence[x++] = curr\_track;

distance = abs(head - curr\_track);

seek\_count += distance;

head = curr\_track;

}

}

int main() {

int i;

printf("\n C-SCAN Disk Scheduling\n");

printf("\n Enter the size of the disk : ");

scanf("%d", & size);

printf("\n Enter the number of tracks to be seeked : ");

scanf("%d", & n);

if (n > MAX) {

printf("\n Number of tracks to be seeked cannot exceed %d Exiting...\n", MAX);

exit(0);

}

printf("\n Enter the starting position of the head : ");

scanf("%d", & head);

if (head > size) {

printf("\n Starting position of head cannot exceed the size of disk. Exiting...\n");

exit(0);

}

printf("\n Enter the tracks to be seeked : ");

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

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

cscands();

printf("\n The Seek Sequence is : ");

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

printf(" %d -> ", sequence[i]);

printf(" %d\n", sequence[i]);

printf("\n The Seek Count is : %d\n", seek\_count);

return 0;

}

**OUTPUT:**

