10.Develop a C program to simulate SCAN disk scheduling algorithms

PROGRAM

```
#include <stdio.h>
#include <stdlib.h> // For abs() function

int main() {
    int t[20], h, i, j, n, temp;
    printf("Enter the number of tracks to be traversed: ");
    scanf("%d", &n);
    printf("Enter the position of head: ");
    scanf("%d", &h);
    printf("Enter the tracks: ");

    for (i = 0; i < n; i++) {
        scanf("%d", &t[i]);
    }

    // Add head position as part of the tracks to simplify calculations
    t[n] = h;
    n++; // Increment n as head position is now part of the tracks

    // Sorting the tracks including the head position
    for (i = 0; i < n; i++) {
        for (j = 0; j < n - i - 1; j++) {
            if (t[j] > t[j + 1]) {
                temp = t[j];
                t[j] = t[j + 1];
                t[j + 1] = temp;
            }
        }
    }

    // Find the index of the head in the sorted array
    int headIndex;
    for (i = 0; i < n; i++) {
        if (t[i] == h) {
            headIndex = i;
            break;
        }
    }

    int totalMovement = 0;
    // Calculate distance moving left towards 0, then right to the end
    if (headIndex != 0) {
        totalMovement += h - t[0]; // Move left to the first track
    }
    if (headIndex != n - 1) {
        totalMovement += t[n - 1] - t[0]; // Move right to the last track
```

```
    }

    printf("\nTotal head movements: %d\n", totalMovement);

    // Calculate average based on movements, not number of requests
    // Note: This might not be a meaningful metric for SCAN, usually total movement is of
interest
    printf("Average head movements: %f\n", totalMovement / (float)(n - 1));

    return 0;
}
```

OUTPUT

```
Enter the number of tracks to be traversed: 5
Enter the position of head: 6
Enter the tracks: 2 3 7 8 9 4

Total head movements: 11
Average head movements: 2.200000
```

Programs:

1. Write a program to handle dynamic arrival of Disk Requests?

2. Write a program to simulate C-SCAN or LOOK algorithm?