(39)

#include <stdio.h>

#include <stdlib.h>

#define MAX 100

void csan(int arr[], int n, int head, int direction) {

int seek\_sequence[MAX], distance, seek\_count = 0, cur\_track;

int i, j;

// Sort the request array

for (i = 0; i < n - 1; i++) {

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

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

int temp = arr[j];

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

arr[j + 1] = temp;

}

}

}

// Add the head to the seek sequence

seek\_sequence[0] = head;

seek\_count++;

// Run the C-SCAN algorithm

if (direction == 1) {

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

if (arr[i] >= head) {

seek\_sequence[seek\_count++] = arr[i];

}

}

seek\_sequence[seek\_count++] = 0; // Jump to the start

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

if (arr[i] < head) {

seek\_sequence[seek\_count++] = arr[i];

}

}

} else {

for (i = n - 1; i >= 0; i--) {

if (arr[i] <= head) {

seek\_sequence[seek\_count++] = arr[i];

}

}

seek\_sequence[seek\_count++] = MAX - 1; // Jump to the end

for (i = n - 1; i >= 0; i--) {

if (arr[i] > head) {

seek\_sequence[seek\_count++] = arr[i];

}

}

}

// Calculate total seek time

for (i = 0; i < seek\_count - 1; i++) {

cur\_track = seek\_sequence[i];

distance = abs(cur\_track - seek\_sequence[i + 1]);

seek\_count += distance;

}

printf("Total seek time: %d\n", seek\_count);

printf("Seek Sequence: ");

for (i = 0; i < seek\_count; i++) {

printf("%d ", seek\_sequence[i]);

}

printf("\n");

}

int main() {

int arr[] = { 176, 79, 34, 60, 92, 11, 41, 114 };

int head = 50;

int direction = 1; // 1 for upward, 0 for downward

int n = sizeof(arr) / sizeof(arr[0]);

csan(arr, n, head, direction);

return 0;

}

