ASSIGNMENT 12

#include <stdio.h>

#include <stdlib.h>

#include <limits.h>

#define NUM\_REQUESTS 5

// Helper function for absolute value

int abs\_diff(int a, int b) {

return (a > b) ? a - b : b - a;

}

// FCFS Calculation

int calculate\_fcfs(int head, int queue[]) {

int movement = 0;

int current = head;

for (int i = 0; i < NUM\_REQUESTS; i++) {

movement += abs\_diff(current, queue[i]);

current = queue[i];

}

return movement;

}

// SSTF Calculation

int calculate\_sstf(int head, int queue[]) {

int visited[NUM\_REQUESTS] = {0};

int movement = 0;

int current = head;

for (int i = 0; i < NUM\_REQUESTS; i++) {

int min\_dist = INT\_MAX;

int index = -1;

for (int j = 0; j < NUM\_REQUESTS; j++) {

if (!visited[j]) {

int dist = abs\_diff(current, queue[j]);

if (dist < min\_dist) {

min\_dist = dist;

index = j;

}

}

}

movement += min\_dist;

current = queue[index];

visited[index] = 1;

}

return movement;

}

int main() {

int head = 55;

int queue[NUM\_REQUESTS] = {10, 70, 75, 23, 65};

int fcfs\_movement = calculate\_fcfs(head, queue);

int sstf\_movement = calculate\_sstf(head, queue);

printf("Initial Head Position: %d\n", head);

printf("Disk Request Queue: ");

for (int i = 0; i < NUM\_REQUESTS; i++) printf("%d ", queue[i]);

printf("\n\n=== Disk Scheduling Comparison ===\n");

printf("FCFS Total Head Movement: %d tracks\n", fcfs\_movement);

printf("SSTF Total Head Movement: %d tracks\n", sstf\_movement);

if (fcfs\_movement > sstf\_movement)

printf("✅ SSTF requires less head movement.\n");

else

printf("✅ FCFS requires less head movement.\n");

return 0;

}