Реализовал алгоритм лифта на C#...

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace \_4.\_2

{

class Program

{

static int size = 10;

static int diskSize = 10;

/// <summary>

/// Implementation of elevator algo with choosing the side of moving

/// </summary>

/// <param name="arr">Arr of elements to visit</param>

/// <param name="head">Start position</param>

/// <param name="direction"> True - left, false - right</param>

static void ElevatorAlgo(int[] arr, int head, bool direction)

{

int headeReserve = head;

int seekCount = 0;

int distance, curTrack;

List<int> left = new List<int>(), right = new List<int>();

List<int> seekSequence = new List<int>();

if (direction)

left.Add(0);

else

right.Add(diskSize - 1);

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

{

if (arr[i] < head)

left.Add(arr[i]);

if (arr[i] > head)

right.Add(arr[i]);

}

left.Sort();

right.Sort();

int run = 2;

while (run-- > 0)

{

if (direction)

{

for (int i = left.Count - 1; i >= 0; i--)

{

curTrack = left[i];

seekSequence.Add(curTrack);

distance = Math.Abs(curTrack - head);

seekCount += distance;

head = curTrack;

}

direction = false;

}

else

{

for (int i = 0; i < right.Count; i++)

{

curTrack = right[i];

seekSequence.Add(curTrack);

distance = Math.Abs(curTrack - head);

seekCount += distance;

head = curTrack;

}

direction = true;

}

}

Console.Write("Seek Sequence is" + "\n");

Console.WriteLine(headeReserve);

for (int i = 0; i < seekSequence.Count; i++)

{

Console.Write(seekSequence[i] + "\n");

}

}

static void Main(string[] args)

{

Console.WriteLine("Select algorithm:\n\t1. Elevator\n\t2. SSF");

var algo = Console.ReadLine();

switch(algo)

{

case "1":

int[] arr = { 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 };

int head = 3;

ElevatorAlgo(arr, head, true);

break;

case "2":

throw new NotImplementedException();

default:

Console.WriteLine("There is no such algorithm!");

break;

}

Console.ReadLine();

}

}

}