

Q3

Description

在老舊的火車站，您也許會遇到少數僅存的"車箱置換員"。

"車箱置換員"是鐵路部門的員工，主要工作就是重新排列火車車廂。

一旦以最佳順序排列了車廂，所有火車司機要做的就是將車廂逐一卸下即可。

"車箱置換員"源自在鐵路橋附近的車站中執行此任務的第一人。

這座橋並不會垂直打開，而是繞著河中央的一根支柱旋轉。將橋旋轉90度後，船隻就能向左或向右駛過。

第一位"車箱置換員"發現，這座橋最多可以在其上運行兩個車廂，通過將橋旋轉180度，車廂就能切換位置。

(缺點是車廂面向相反的方向，但是火車車廂可以以任何一種方式移動，所以沒差)。

現在幾乎所有的"車箱置換員"都已經淘汰了，鐵路公司希望將其操作自動化。

你的任務就是寫一個程式，該程式要計算最少需要交換幾次兩個相鄰車廂，才能將所有車廂依序排好。

Input

輸入的第一行包含一個整數N，N代表測資數量。

每組測資的第一行包含一個整數L ($0 \leq L \leq 50$)，L代表火車的長度。

第二行包含數字1到L的排列，表示火車車廂的當前順序。

需要將火車車廂依照編號1到L的順序排好。

Output

對於每組測資，請輸出：

"Optimal train swapping takes S swaps."，S代表最少交換次數。

Sample Input 1

```
3
3
1 3 2
4
4 3 2 1
2
2 1
```

Sample Output 1

```
Optimal train swapping takes
1 swaps.
Optimal train swapping takes
6 swaps.
Optimal train swapping takes
1 swaps.
```

Problems

Announcements

Submissions

Rankings

View Contest

Information

ID 122803

Time Limit 3000MS

Memory Limit 512MB

IO Mode Standard IO

Created By m0968390

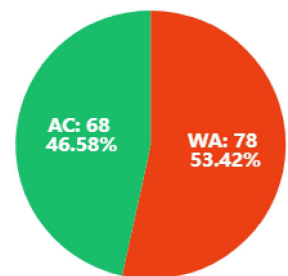
Level Low

Tags [Show](#)

Statistic

Details

AC WA



題目大意：

將每筆測資之數列由小到大排序，並計算交換次數。

(相當於Bubble Sort，一次比較兩兩相鄰元素，若前者較大則交換，一輪結束後，最大元素會落在陣列尾端)

用迴圈比較兩兩相鄰元素，若前者較大則交換，並將交換次數++

最後輸出總交換次數

Language:

C



Theme:

Solarized Light



```
1 #include<stdio.h>
2
3 int main(){
4
5     int N;
6     int L;
7     int train[50] = {0};
8     int i, j;
9     int temp;
10
11     scanf("%d", &N);           //number of case
12
13     while(N--){
14
15         scanf("%d", &L);       //length of train
16
17         int count = 0;
18
19         for(i = 0; i < L; i++){
20             scanf("%d", &train[i]);
21         }
22
23         for(i = 0; i < L - 1; i++){
24             for(j = 0; j < L - 1 - i; j++){
25                 if(train[j] > train[j + 1]){
26                     temp = train[j];
27                     train[j] = train[j + 1];
28                     train[j + 1] = temp;
29                     count++;
30                 }
31             }
32         }
33
34         printf("Optimal train swapping takes %d swaps.\n", count);
35
36     }
37
38     return 0;
39 }
```

逢甲大學線上解題系統

Powered by [OnlineJudge](#) Version: 20190627-b3b6a