



Online Judge	Problem Set	Authors	Online Contests	User
Web Board Home Page F.A.Qs Statistical Charts	Problems Submit Problem Online Status Prob.ID: <input type="text"/> <input type="button" value="Go"/>	Register Update your info Authors ranklist <input type="text"/> <input type="button" value="Search"/>	Current Contest Past Contests Scheduled Contests Award Contest	User ID: <input type="text"/> Password: <input type="password"/> <input type="button" value="login"/> Register

Ultra-QuickSort

Language: ▾

Time Limit: 7000MS **Memory Limit:** 65536K
Total Submissions: 63314 **Accepted:** 23621

Description

In this problem, you have to analyze a particular sorting algorithm. The algorithm processes a sequence of n distinct integers by swapping two adjacent sequence elements until the sequence is sorted in ascending order. For the input sequence

9 1 0 5 4 ,

Ultra-QuickSort produces the output

0 1 4 5 9 .

Your task is to determine how many swap operations Ultra-QuickSort needs to perform in order to sort a given input sequence.

Input

The input contains several test cases. Every test case begins with a line that contains a single integer $n < 500,000$ -- the length of the input sequence. Each of the the following n lines contains a single integer $0 \leq a[i] \leq 999,999,999$, the i -th input sequence element. Input is terminated by a sequence of length $n = 0$. This sequence must not be processed.

Output



For every input sequence, your program prints a single line containing an integer number op, the minimum number of swap operations necessary to sort the given input sequence.

Sample Input

5
9
1
0
5
4
3
1
2
3
0

Sample Output

6
0

Source

Waterloo local 2005.02.05

[\[Submit\]](#) [\[Go Back\]](#) [\[Status\]](#) [\[Discuss\]](#)



[Home Page](#)



[Go Back](#)



[To top](#)