

# P4375 [USACO18OPEN] Out of Sorts G

## 题目描述

留意着农场之外的长期职业生涯的可能性，奶牛 Bessie 开始在不同的在线编程网站上学习算法。

她到目前为止最喜欢的算法是“冒泡排序”。以下是 Bessie 最初的对长度为  $N$  的数组  $A$  进行排序的代码实现：

```
sorted = false
while (not sorted):
    sorted = true
    moo
    for i = 0 to N-2:
        if A[i+1] < A[i]:
            swap A[i], A[i+1]
        sorted = false
```

显然，代码中的 `moo` 指令的作用只是输出“moo”。奇怪的是，Bessie 似乎执着于在她的代码中的不同位置使用这个语句。

在用若干个数组测试了她的代码之后，Bessie 观察到一个有趣的现象：大的元素很快就会被拉到数组末尾，而小的元素需要很长时间“冒泡”到数组的开头（她怀疑这就是这个算法得名的原因）。为了实验和缓解这一问题，Bessie 修改了她的代码，使代码在每次循环中向前再向后各扫描一次，从而无论是大的元素还是小的元素在每一次循环中都有机会被拉较长的一段距离。她的代码现在是这样的：

```
sorted = false
while (not sorted):
    sorted = true
    moo
    for i = 0 to N-2:
        if A[i+1] < A[i]:
            swap A[i], A[i+1]
    for i = N-2 downto 0:
        if A[i+1] < A[i]:
            swap A[i], A[i+1]
    for i = 0 to N-2:
        if A[i+1] < A[i]:
            sorted = false
```

给定一个输入数组，请预测 Bessie 修改后的代码会输出多少次“moo”。

## 输入格式

输入的第一行包含  $N$  ( $1 \leq N \leq 100,000$ )。接下来的  $N$  行描述了  $A[0] \dots A[N-1]$ ，每个数都是一个范围在  $0 \dots 10^9$  的整数。输入数据不保证各不相同。

## 输出格式

输出“moo”被输出的次数。

## 输入输出样例 #1

### 输入 #1

```
5
1
8
5
3
2
```

# 输出 #1

2

# 说明/提示

供题：Brian Dean