

B. 一排再排

Description

既然你會寫 C 語言，那你會寫插入排序法 (insertion sort) 嗎？插入排序法是一個很簡單的演算法，用來把給定的一個序列 L 進行排序 (先假設都是整數然後要由小排到大吧！)，流程如下：

1. 從 L 裡的第一個元素開始依序進行下面的操作。
2. 如果當前元素已經是序列裡面的第一個元素，或者當前元素的前一個位置的元素沒有比當前元素大，則停止對當前元素的操作。
3. 否則將當前元素和前一個位置的元素交換。
4. 回到步驟 2。

看不懂，或者覺得很神奇嗎？你可以自己上網查詢一下「插入排序法」，或者是看一下下面的例子來心領神會。假設初始的序列是 $[3\ 2\ 4\ 1]$ ，那麼插入排序的過程大概像是：

1. 首先我們來看第一個元素 3
2. 3 已經是序列的第一個元素了，所以結束
3. 再來我們看第二個元素 2
4. 2 前一個位置的元素 3 比 2 還大，所以把 2 和 3 位置交換
5. 2 現在是序列的第一個元素了，所以結束
6. 然後我們看第三個元素 4
7. 4 前一個位置的元素 3 沒有比 4 大，所以結束
8. 最後來看第四個元素 1
9. 1 前一個位置的元素 4 比 1 大，所以把 1 和 4 位置交換
10. 1 前一個位置的元素 3 比 1 大，所以把 1 和 3 位置交換
11. 1 前一個位置的元素 2 比 1 大，所以把 1 和 2 位置交換
12. 1 現在是序列的第一個元素了，所以結束

然後我們就會發現，序列已經被排好了，真的是太厲害了。事實上，插入排序的特性是，假如前面 k 個元素已經被處理好了，那麼被更動過後的當前序列的前 k 個元素就會是排序好的狀態。

阿部第一次學習插入排序法就被這項魔法一般的力量震懾住了。「阿... 這個插入帶來的震撼... 哦...」在阿部第一次做完上面的例子後，他覺得意猶未盡，於是他定義了一個新的序列，叫做「位移序列」。對於一個序列 L ，阿部定義對應的位移序列 $f(L)$ 為一個長度和 L 一樣，且第 i 個元素為在插入排序法對 L 中的第 i 個元素進行操作時，往前交換的次數。舉例而言，在上面的例子 $L = [3\ 2\ 4\ 1]$ 中，3 沒有跟前面的任何元素交換、2 跟 3 交換一次、4 沒有跟任何元素交換、1 和所有其它元素都交換了一次，所以 $f(L) = [0\ 1\ 0\ 3]$ 。有了位移序列的定義後，阿部就可以好好享受插入排序法帶來的樂趣了：對於一個序列 L ，他會先做一次插入排序法，獲得位移序列 $f(L)$ ，然後再對這個位移序列做插入排序法，獲得位移序列 $f(f(L))$ ，一直重複做下去，這樣就可以一直維持插入元素的快感。

但是，很快他發現，這樣子一定有一天獲得的最新的位移序列會全部都是 0 (想一想，為什麼?)——這樣子一看就知道下一回合獲得的位移序列依舊全部都是 0，而且從此以後就不會再有變化，這樣子插入排序就沒有意義，事情就變得不有趣了。所以，他據此對一個序列定義了「有趣度」。一個序列 L 的有趣度為「多少輪插入排序後計算出來的位移序列會全部都是 0」。在上面的例子中， $L = [3\ 2\ 4\ 1]$ ， $f(L) = [0\ 1\ 0\ 3]$ ， $f(f(L)) = [0\ 0\ 1\ 0]$ ， $f(f(f(L))) = [0\ 0\ 0\ 1]$ ， $f(f(f(f(L)))) = [0\ 0\ 0\ 0]$ ，做了 4 輪之後獲得的位移序列就全部都是 0 了，所以 L 的有趣度為 4。

你是阿部有一天在公園裡無意間認識的好朋友，你很喜歡跟阿部一起玩，但是最近你買了 PS4，在上面玩巫師 3，而那是款單人遊戲，兩個人玩很容易起爭執。為此，你準備了很多序列好讓阿部把心思花在練習插入排序法上。顯然，有趣度越高的序列能讓阿部玩越久，這樣你就能霸佔更久的機器，因此你決定寫一個程式計算出所有你準備的序列的有趣度，好讓你可以根據你想玩的時間選擇要給阿部哪個序列。

Input

輸入的第一行是一個正整數 T ，代表有多少個序列需要計算。接著共有 T 行，每行描述一個序列。第一個數值 n ，代表當前序列的長度，剩下的 n 個數值 a_1, a_2, \dots, a_n 依序代表序列的元素。

- 對於 20% 的測試資料， $n \leq 3$ 。
- 對於 40% 的測試資料， $n \leq 4$ 。
- 對於 100% 的測試資料， $1 \leq T \leq 100, 1 \leq n \leq 100, 1 \leq a_i \leq 1000$ ，其中 a_i 代表任意序列中的任意元素。

Output

對於每一個序列，輸出該序列的有趣度。

Sample

Input	Output
4	4
4 3 2 4 1	1
1 888	2
3 3 2 1	1
5 5 5 5 5 5	

Hint

1. 你會需要熟悉迴圈和陣列操作。
2. 你可能會使用到超過一個陣列。