Problem J
John's Book Stack

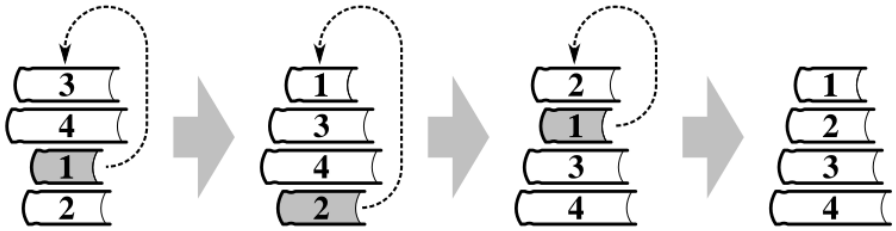
Problem ID: johnsstack
CPU Time limit: 1 second
Memory limit: 1024 MB

Source: Benelux Algorithm Programming Contest (BAPC) 2012
License: CC BY-SA

John has a big stack of books of various sizes. Such a stack is stable if the books have non-decreasing sizes (viewed from top to bottom); otherwise, it is unstable, and likely to fall over.

To prevent this, John wants to sort the books in the stack by size. He does so by pulling out a book from somewhere in the middle (or bottom) of the stack and then putting it back on top. However, he can only pull out a book safely if the books on top of it already form a stable stack.

For example, if John has a stack of four books with sizes 3, 4, 1 and 2 (from top to bottom) then he can sort them as follows:



Your task is to determine how many steps are required to sort a given stack of books. In the example above, which corresponds to the first sample case, the answer is 3.

Input

On the first line one positive number: the number of test cases, at most 100. After that per test case:

- one line with an integer n ($1 \leq n \leq 50$): the number of books.
- one line containing n space-separated integers s_i ($1 \leq s_i \leq 1\,000$ for $1 \leq i \leq n$): the sizes of the books, as they appear in the initial stack from top to bottom.

Output

Per test case:

- one line with an integer: the minimum number of steps required to sort the stack using the algorithm described above.

Sample Input 1

```
4
4
3 4 1 2
8
3 1 4 1 5 9 2 6
5
1 42 42 42 1000
22
4 1 2 5 6 7 9 10 3 13 17 11 12 14 19 20 22 8 15 16 18 21
```

Sample Output 1

```
3
53
0
1234567
```