Brick Comparisions

Chef has N bricks with him, numbered 1 to N, with the i-th one being of size A_i.

He will perform the following process:

- First, he picks up the brick numbered 1.
- Next, he looks at the other bricks in the order $2, 3, \ldots, N$, and if he finds this brick has a **strictly larger** size than the brick in his hand, he drops his current brick and picks this up instead.

Your task is to find the number of the brick which is finally with Chef at the end of the process.

For example, when $A = [2, 1, 4, 3]$. The process happens as follows:

- Chef first picks up the brick numbered 1.
- Chef looks at brick numbered 2, but it does not have a larger size than his current brick (which is brick 1)
- Chef looks at brick numbered 3, and it has a larger size than his current brick. Hence, he drops the brick numbered 1 and picks up the brick numbered 3.
- Chef looks at brick numbered 4, but it does not have a larger size than his current brick (which is brick 3)

Hence, the final brick with Chef is the brick numbered 3.

Input Format

- The first line of input will contain a single integer T, denoting the number of test cases.
- Each test case consists of multiple lines of input.
  - The first line of each test case contains N - the number of bricks.
  - The second line contains N integers - $A_1, A_2, \ldots, A_N$.

Output Format

For each test case, output on a new line the number of the brick which is finally with Chef.

Constraints

- $1 \leq T \leq 100$
- $1 \leq N \leq 100$
- $1 \leq A_i \leq 100$

Sample 1:

Input	Output

3
4
2 1 4 3
2
	3
	1
	3