

Digesting

Input file: **standard input**
Output file: **standard output**
Time limit: 1 second
Memory limit: 1024 megabytes

Let's define *digesting* of an array $a_1, a_2, \dots, a_n$ the following way:

- Choose any permutation of length n : $p_1, p_2, \dots, p_n$ (an array consisting of integers from 1 to n , such that each of them appears exactly once).
- Simultaneously perform $a_i := a_i \bmod \max(a_{p_i}, 1)$.
- Choose any number a_i to kick from the array, concatenate others into **new** array $a_1, a_2, \dots, a_{n-1}$. This array is the result of the digesting operation.

Any array that can be obtained by this procedure, can be the result of digesting operation.

You are given an array $a_1, a_2, \dots, a_n$ consisting of integers from 1 to n . What is the maximum possible value of a_1 that can be achieved, after digesting this array $n - 1$ times in a row?

Input

Each test consists of multiple test cases. The first line contains an integer t ($1 \leq t \leq 10^4$) — the number of test cases. The following lines describe the test cases.

The first line of each test case contains an integer n ($1 \leq n \leq 2 \cdot 10^5$) — the length of the array a .

The second line of each test case contains n integers $a_1, a_2, \dots, a_n$ ($1 \leq a_i \leq n$) — the elements of the array a .

It is guaranteed that the sum of the values of n for all test cases does not exceed $2 \cdot 10^5$.

Output

For each test case, output the maximum possible value of a_1 that can be achieved.

Example

standard input	standard output
1 3 3 2 2	1