

A. DIY Wooden Ladder

time limit per test: 2 seconds

memory limit per test: 256 megabytes

input: standard input

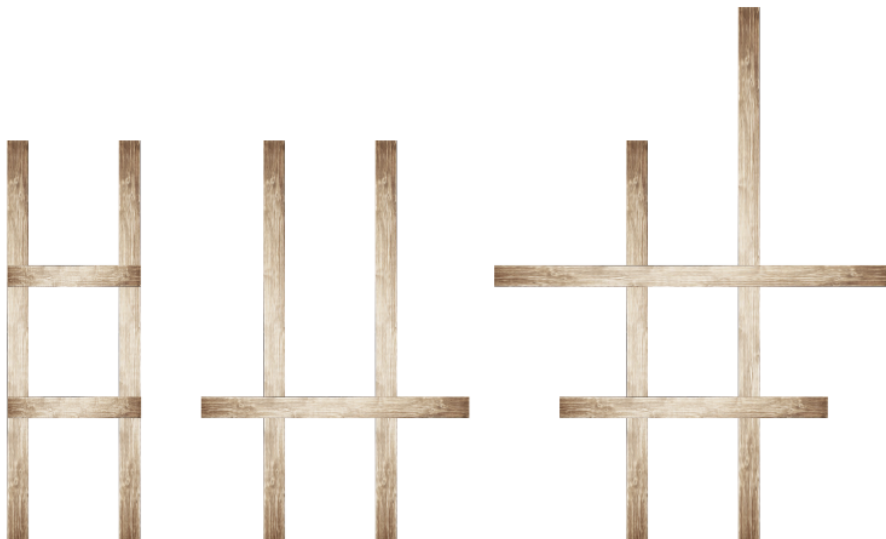
output: standard output

Let's denote a k -step ladder as the following structure: exactly $k + 2$ wooden planks, of which

- two planks of length **at least** $k + 1$ — the base of the ladder;
- k planks of length **at least** 1 — the steps of the ladder;

Note that neither the base planks, nor the steps planks are required to be equal.

For example, ladders 1 and 3 are correct 2-step ladders and ladder 2 is a correct 1-step ladder. On the first picture the lengths of planks are $[3, 3]$ for the base and $[1]$ for the step. On the second picture lengths are $[3, 3]$ for the base and $[2]$ for the step. On the third picture lengths are $[3, 4]$ for the base and $[2, 3]$ for the steps.



You have n planks. The length of the i -th planks is a_i . You don't have a saw, so you can't cut the planks you have. Though you have a hammer and nails, so you can assemble the improvised "ladder" from the planks.

The question is: what is the maximum number k such that you can choose some subset of the given planks and assemble a k -step ladder using them?

Input

The first line contains a single integer T ($1 \leq T \leq 100$) — the number of queries. The queries are independent.

Educational Codeforces Round 69 (Rated for Div. 2)

Finished

→ Practice?

Want to solve the contest problems after the official contest ends? Just register for practice and you will be able to submit solutions.

[Register for practice](#)

→ Virtual participation

Virtual contest is a way to take part in past contest, as close as possible to participation on time. It is supported only ACM-ICPC mode for virtual contests. If you've seen these problems, a virtual contest is not for you - solve these problems in the archive. If you just want to solve some problem from a contest, a virtual contest is not for you - solve this problem in the archive. Never use someone else's code, read the tutorials or communicate with other person during a virtual contest.

[Start virtual contest](#)

→ Problem tags

[greedy](#) [math](#) [sortings](#) [*1000](#)

No tag edit access

→ Contest materials

- [Announcement \(en\)](#) [×](#)
- [Tutorial #1 \(en\)](#) [×](#)
- [Tutorial #2 \(en\)](#) [×](#)
- [Tutorial #3 \(ru\)](#) [×](#)

Each query consists of two lines. The first line contains a single integer n ($2 \leq n \leq 10^5$) — the number of planks you have.

The second line contains n integers a_1, a_2, \dots, a_n ($1 \leq a_i \leq 10^5$) — the lengths of the corresponding planks.

It's guaranteed that the total number of planks from all queries doesn't exceed 10^5 .

Output

Print T integers — one per query. The i -th integer is the maximum number k , such that you can choose some subset of the planks given in the i -th query and assemble a k -step ladder using them.

Print 0 if you can't make even 1-step ladder from the given set of planks.

Example

input	Copy
4 4 1 3 1 3 3 3 3 2 5 2 3 3 4 2 3 1 1 2	
output	Copy
2 1 2 0	

Note

Examples for the queries 1 — 3 are shown at the image in the legend section.

The Russian meme to express the quality of the ladders:

