

🎁 Codeforces celebrates 10 years! We are pleased to announce the crowdfunding-campaign. Congratulate us by the link <https://codeforces.com/10years>.

C. Mind Control

time limit per test: 1 second

memory limit per test: 256 megabytes

input: standard input

output: standard output

You and your $n - 1$ friends have found an array of integers a_1, a_2, \dots, a_n . You have decided to share it in the following way: All n of you stand in a line in a particular order. Each minute, the person at the front of the line chooses either the first or the last element of the array, removes it, and keeps it for himself. He then gets out of line, and the next person in line continues the process.

You are standing in the m -th position in the line. **Before the process starts**, you may choose up to k different people in the line, and persuade them to always take either the first or the last element in the array on their turn (for each person his own choice, not necessarily equal for all people), no matter what the elements themselves are. **Once the process starts, you cannot persuade any more people, and you cannot change the choices for the people you already persuaded.**

Suppose that you're doing your choices optimally. What is the greatest integer x such that, no matter what are the choices of the friends you didn't choose to control, the element you will take from the array will be greater than or equal to x ?

Please note that the friends you don't control may do their choice arbitrarily, and they will not necessarily take the biggest element available.

Input

Processing math: 100%

Codeforces Round #616 (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 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

The input consists of multiple test cases. The first line contains a single integer t ($1 \leq t \leq 1000$) — the number of test cases. The description of the test cases follows.

The first line of each test case contains three space-separated integers n , m and k ($1 \leq m \leq n \leq 3500$, $0 \leq k \leq n - 1$) — the number of elements in the array, your position in line and the number of people whose choices you can fix.

The second line of each test case contains n positive integers a_1, a_2, \dots, a_n ($1 \leq a_i \leq 10^9$) — elements of the array.

It is guaranteed that the sum of n over all test cases does not exceed 3500.

Output

For each test case, print the largest integer x such that you can guarantee to obtain at least x .

Example

input

Copy

```
4
6 4 2
2 9 2 3 8 5
4 4 1
2 13 60 4
4 1 3
1 2 2 1
2 2 0
1 2
```

output

Copy

```
8
4
1
1
```

Note

In the first test case, an optimal strategy is to force the first person to take the last element and the second person to take the first element.

- the first person will take the last element (5) because he or she was forced by you to take the last element. After this turn the remaining array will be [2, 9, 2, 3, 8];
- the second person will take the first element (2) because he or she was forced by you to take the first element. After this turn the remaining array will be [9, 2, 3, 8];
- if the third person will choose to take the first element (9), at your turn the remaining array will be [2, 3, 8] and you will take 8 (the last element);

→ Problem tags

brute force

data structures


greedy

math

*1700

No tag edit access

→ Contest materials

- Announcement (en) 

- Tutorial (en) 

- if the third person will choose to take the last element (8), at your turn the remaining array will be [9, 2, 3] and you will take 9 (the first element).

Thus, this strategy guarantees to end up with at least 8. We can prove that there is no strategy that guarantees to end up with at least 9. Hence, the answer is 8.

In the second test case, an optimal strategy is to force the first person to take the first element. Then, in the worst case, both the second and the third person will take the first element: you will end up with 4.

[Codeforces](#) (c) Copyright 2010-2020 Mike Mirzayanov
The only programming contests Web 2.0 platform
Server time: Feb/05/2020 23:25:17^{UTC+1} (e2).
Desktop version, switch to [mobile version](#).
[Privacy Policy](#).

Supported by



ITMO UNIVERSITY