

## C. To Become Max

time limit per test: 2 seconds  
memory limit per test: 256 megabytes

You are given an array of integers  $a$  of length  $n$ .

In one operation you:

- Choose an index  $i$  such that  $1 \leq i \leq n - 1$  and  $a_i \leq a_{i+1}$ .
- Increase  $a_i$  by 1.

Find the maximum possible value of  $\max(a_1, a_2, \dots, a_n)$  that you can get after performing this operation at most  $k$  times.

### Input

Each test contains multiple test cases. The first line of input contains a single integer  $t$  ( $1 \leq t \leq 100$ ) — the number of test cases. The description of the test cases follows.

The first line of each test case contains two integers  $n$  and  $k$  ( $2 \leq n \leq 1000$ ,  $1 \leq k \leq 10^8$ ) — the length of the array  $a$  and the maximum number of operations that can be performed.

The second line of each test case contains  $n$  integers  $a_1, a_2, \dots, a_n$  ( $1 \leq a_i \leq 10^8$ ) — the elements of the array  $a$ .

It is guaranteed that the sum of  $n$  over all test cases does not exceed 1000.

### Output

For each test case output a single integer — the maximum possible maximum of the array after performing at most  $k$  operations.

### Example

input	Copy
6	
3 4	
1 3 3	
5 6	
1 3 4 5 1	
4 13	
1 1 3 179	
5 3	
4 3 2 2 2	
5 6	
6 5 4 1 5	
2 17	
3 5	
output	Copy
4	
7	
179	
5	
7	
6	

### Note

In the first test case, one possible optimal sequence of operations is:

$[1, 3, 3] \rightarrow [2, 3, 3] \rightarrow [2, 4, 3] \rightarrow [3, 4, 3] \rightarrow [4, 4, 3]$ .

In the second test case, one possible optimal sequence of operations is:

$[1, 3, 4, 5, 1] \rightarrow [1, 4, 4, 5, 1] \rightarrow [1, 5, 4, 5, 1] \rightarrow [1, 5, 5, 5, 1] \rightarrow [1, 5, 6, 5, 1] \rightarrow [1, 6, 6, 5, 1] \rightarrow [1, 7, 6, 5, 1]$

**Codeforces Round 890 (Div. 2)**  
**supported by Constructor**  
**Institute**

**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](#)

### → Problem tags

[binary search](#) [brute force](#)  
[data structures](#) [dp](#) [\\*1600](#)

No tag edit access

### → Contest materials

- [Announcement \(en\)](#) ✕
- [Tutorial \(en\)](#) ✕

The only programming contests Web 2.0 platform  
Server time: Jan/04/2025 12:53:50<sup>UTC+9</sup> (j3).  
Desktop version, switch to [mobile version](#).  
[Privacy Policy](#).

Supported by



**ITMO**