

## B. Maximum of Maximums of Minimums

time limit per test: 1 second  
memory limit per test: 256 megabytes  
input: standard input  
output: standard output

You are given an array  $a_1, a_2, \dots, a_n$  consisting of  $n$  integers, and an integer  $k$ . You have to split the array into exactly  $k$  non-empty subsegments. You'll then compute the minimum integer on each subsegment, and take the maximum integer over the  $k$  obtained minimums. What is the maximum possible integer you can get?

Definitions of subsegment and array splitting are given in notes.

### Input

The first line contains two integers  $n$  and  $k$  ( $1 \leq k \leq n \leq 10^5$ ) — the size of the array  $a$  and the number of subsegments you have to split the array to.

The second line contains  $n$  integers  $a_1, a_2, \dots, a_n$  ( $-10^9 \leq a_i \leq 10^9$ ).

### Output

Print single integer — the maximum possible integer you can get if you split the array into  $k$  non-empty subsegments and take maximum of minimums on the subsegments.

### Examples

<b>input</b>	<a href="#">Copy</a>
5 2 1 2 3 4 5	
<b>output</b>	<a href="#">Copy</a>
5	

  

<b>input</b>	<a href="#">Copy</a>
5 1 -4 -5 -3 -2 -1	
<b>output</b>	<a href="#">Copy</a>
-5	

### Note

A subsegment  $[l, r]$  ( $l \leq r$ ) of array  $a$  is the sequence  $a_l, a_{l+1}, \dots, a_r$ .

Splitting of array  $a$  of  $n$  elements into  $k$  subsegments  $[l_1, r_1], [l_2, r_2], \dots, [l_k, r_k]$  ( $l_1 = 1, r_k = n, l_i = r_{i-1} + 1$  for all  $i > 1$ ) is  $k$  sequences  $(a_{l_1}, \dots, a_{r_1}), \dots, (a_{l_k}, \dots, a_{r_k})$ .

In the first example you should split the array into subsegments  $[1, 4]$  and  $[5, 5]$  that results in sequences  $(1, 2, 3, 4)$  and  $(5)$ . The minimums are  $\min(1, 2, 3, 4) = 1$  and  $\min(5) = 5$ . The resulting maximum is  $\max(1, 5) = 5$ . It is obvious that you can't reach greater result.

In the second example the only option you have is to split the array into one subsegment  $[1, 5]$ , that results in one sequence  $(-4, -5, -3, -2, -1)$ . The only minimum is  $\min(-4, -5, -3, -2, -1) = -5$ . The resulting maximum is  $-5$ .

### Technocup 2018 - Elimination Round 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](#)

### → Problem tags

[greedy](#) [\\*1200](#)

No tag edit access

### → Contest materials

- Announcement
- Tutorial

---

[Codeforces](#) (c) Copyright 2010-2020 Mike Mirzayanov  
The only programming contests Web 2.0 platform  
Server time: Sep/22/2020 15:22:22<sup>UTC-5</sup> (i1).  
Desktop version, switch to [mobile version](#).  
[Privacy Policy](#)

Supported by



ITMO UNIVERSITY