

ACM-ICPC Indonesia National Contest 2016

Problem H

Negotiation Game

Time Limit: 2 seconds

Adi and Irvan is a best friends duo. They play, they eat, they joke, they smile, and they fight each other (note: it's a best friend kind of fight). They are also phd candidates (computer science), so, of course they discuss their research together. They often solve their disputes by their so-called "smart" negotiation.

One day, Adi and Irvan are faced with an array of N integer, $A[1..N]$. Looking at this array, Adi and Irvan tempted to pick one element to bring home. However, together they can only choose exactly one element only to bring home. Problem arises, when they disagree on the element which should be chosen. Adi wants the value to be as large as possible, while Irvan wants the value to be as small as possible.

Being an eccentric person, Irvan suggests an unconventional way to solve this dispute. Adi and Irvan will take turns doing these procedures (`split_and_keep`) to reduce the size of A while it still has more than 1 element in it. As an older one, Adi moves first.

```

1  split_and_keep(A[1..M])
2      x ← between 1 and M-1, inclusive
3      B ← either A[1..x] or A[x+1..M]
4      A = B

```

Line 2 and 3 in `split_and_keep` pseudocode above is where Adi and Irvan make a decision (recall that their objectives are different). Notice that every call of `split_and_keep` will reduce the size of A . The only element remains in A at the end will be their chosen element.

If Adi and Irvan play this negotiation (game) optimally, i.e. Adi tries to minimize the chosen element while Irvan tries to maximize it, what will the value of the chosen element be?

For example, let $A = \{7, 3, 4, 8, 7, 4\}$. One possible game will proceed like this:

$A = 7\ 3\ 4\ 8\ 7\ 4$

Adi : $x = 2$ and pick $A[3..6] = \{4, 8, 7, 4\}$

$A = 4\ 8\ 7\ 4$

Irvan : $x = 2$ and pick $A[1..2] = \{4, 8\}$

$A = 4\ 8$

Adi : $x = 1$ and pick $A[1..1] = \{4\}$

$A = 4$

In this example play, the chosen element's value is 4, which happened to be the chosen value when both play optimally.

Input

The first line of input contains an integer T ($T \leq 100$) denoting the number of cases. Each case begins with an integer N ($1 \leq N \leq 2,000$) denoting the size of array A . The next line contains N integers A_i ($1 \leq A_i \leq 1,000,000$) representing the elements in A , for $i = 1..N$ respectively.

Output

For each case, output in a line "Case #X: Y" where X is the case number, starts from 1, and Y is the output for that particular case.

Sample Input

```
4
6
7 3 4 8 7 4
2
10 20
1
1000
5
5 4 3 2 1
```

Output for Sample Input

```
Case #1: 4
Case #2: 10
Case #3: 1000
Case #4: 1
```