

## 7068 K.Bro Sorting

Matt's friend K.Bro is an ACMer.

Yesterday, K.Bro learnt an algorithm: Bubble sort. Bubble sort will compare each pair of adjacent items and swap them if they are in the wrong order. The process repeats until no swap is needed.

Today, K.Bro comes up with a new algorithm and names it K.Bro Sorting.

There are many rounds in K.Bro Sorting. For each round, K.Bro chooses a number, and keeps swapping it with its next number while the next number is less than it.

For example, if the sequence is "1 4 3 2 5", and K.Bro chooses "4", he will get "1 3 2 4 5" after this round. K.Bro Sorting is similar to Bubble sort, but it's a randomized algorithm because K.Bro will choose a random number at the beginning of each round. K.Bro wants to know that, for a given sequence, how many rounds are needed to sort this sequence in the best situation. In other words, you should answer the minimal number of rounds needed to sort the sequence into ascending order. To simplify the problem, K.Bro promises that the sequence is a permutation of 1, 2, ...,  $N$ .

### Input

The first line contains only one integer  $T$  ( $T \leq 200$ ), which indicates the number of test cases. For each test case, the first line contains an integer  $N$  ( $1 \leq N \leq 10^6$ ).

The second line contains  $N$  integers  $a_i$  ( $1 \leq a_i \leq N$ ), denoting the sequence K.Bro gives you.

The sum of  $N$  in all test cases would not exceed  $3 \times 10^6$ .

### Output

For each test case, output a single line 'Case # $x$ :  $y$ ', where  $x$  is the case number (starting from 1),  $y$  is the minimal number of rounds needed to sort the sequence.

### Hint:

In the second sample, we choose "5" so that after the first round, sequence becomes "1 2 3 4 5", and the algorithm completes.

### Sample Input

```
2
5
5 4 3 2 1
5
5 1 2 3 4
```

### Sample Output

```
Case #1: 4
Case #2: 1
```