Sort a Vector R S98018_en

A **vector R** is a vector that is composed of two parts:

$$v = x_1\ x_2\ x_3\ x_4 \dots x_n\ y_1\ y_2\ y_3 \dots y_m$$

such that the part $x_1 \dots x_n$ and the part $y_1 \dots y_m$ are ordered strictly in increasing order but $y_m < x_1$. Furthermore, we have that $n, m > 0$. That is, neither part is empty.

We need to implement the **function** `void ordena(vector<int>& v, int pos)` with the following specification:

`PRE`: v is a vector R such that $|v| \geq 3$, and pos is the position of y_1 in v.

`POST`: The vector v is sorted.

Observation

You only need to send the function we ask for and the actions and functions that you define yourself. The rest will be ignored.
You cannot use the operation `sort` from the `stl` library.
Hint: knowing the position of y_1 can help you sort the vector in linear time.
On the other hand, if you use an auxiliary vector to sort you will have a penalty of -5 on manual correction even if you have a green traffic light.

Input

An undetermined number of vectors R with the following format: an integer indicating their size, then the vector R and finally the position where y_1 is located. Every vector R has a size greater than or equal to 3.

Output

The sorted vector v.

Sample input

```
15
11 12 13 14 15 1 2 3 4 5 6 7 8 9 10
5

15
3 4 5 6 7 8 9 10 11 12 13 14 15 1 2
13

15
15 1 2 3 4 5 6 7 8 9 10 11 12 13 14
1

5
12 15 4 7 8
2
```

Sample output

```
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
4 7 8 12 15
```

Problem information

Author : PRO1
Generation : 2025-01-10 20:04:31