
Sort a Vector V

Y80100_en

A **vector V** 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 $x_1 \dots x_n$ is ordered in a strictly **decreasing** manner and $y_1 \dots y_m$ is ordered in a strictly **increasing** manner. Furthermore, $x_n > y_1$. Finally, we have that $n, m > 0$. That is, neither part is empty.

We need to implement the **function**

```
void ordena(const vector<int>& v, int pos, vector<int>& r)
```

with the following specification:

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

POST: The vector r contains all the elements of the vector v and 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 not be taken into account.

The operation `sort` from the `std` library cannot be used.

Hint: knowing the position of y_1 can help you sort the vector in linear time.

IMPORTANT: You only need to submit the requested function, and possibly other necessary actions and functions. However, you must keep the type definitions and `#includes`.

Input

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

Output

The vector r is sorted and contains all the elements of v .

Sample input 1

```
10
20 18 16 2 4 6 8 10 12 14
3

10
20 2 4 6 8 10 12 14 16 18
1

10
20 18 16 2 4 6 8 10 12 14
3
```

```
5
14 11 8 2 14
3

10
20 2 4 6 8 10 12 14 16 18
1
```

Sample output 1

```
2 4 6 8 10 12 14 16 18 20
2 4 6 8 10 12 14 16 18 20
```

```
2 4 6 8 10 12 14 16 18 20
2 8 11 14 14
2 4 6 8 10 12 14 16 18 20
```

Problem information

Author: PRO1

Generation: 2026-01-25T23:07:20.839Z

© *Jutge.org*, 2006–2026.

<https://jutge.org>