

Swapsies

Kevin and Daniel are playing a simple game involving sorting. They take turns in giving each other a sequence of real numbers. The other person needs to work out which positions should be swapped in order to have the numbers sorted in descending order.

Input

The first line of the input file will contain N , the number of items for you to sort, $0 \leq N \leq 1000$. Following that will be N lines each containing a real number m ($-1000 \leq m \leq 1000$).

Output

The output should consist of at most N lines containing pairs of integers ab separated by spaces. Each line indicates that the a -th number in the sequence is being swapped with the b -th ($1 \leq a, b \leq N$). Following these swaps should be a single line containing the phrase **all sorted**.

After performing all of the swaps listed in the output, the sequence should be sorted in descending order. For example, given a sequence of $\{-2, 3, -5, 7, 1, -9\}$ as input, the sequence should be $\{7, 3, 1, -2, -5, -9\}$ after all of the swaps are carried out.

Sample Input 1

```
10
5
-8
-2
-9
3
-3
2
1
17
-6
```

Sample Output 1

```
1 9
2 9
3 5
4 7
5 8
6 8
7 8
8 10
all sorted
```

This input file describes a sequence of numbers $\{5, -8, -2, -9, 3, -3, 2, 1, 17, -6\}$. Carrying out the sequence of swaps sorts the data as shown below:

start	$\{5, -8, -2, -9, 3, -3, 2, 1, 17, -6\}$
(1,9)	$\{17, -8, -2, -9, 3, -3, 2, 1, 5, -6\}$
(2,9)	$\{17, 5, -2, -9, 3, -3, 2, 1, -8, -6\}$
(3,5)	$\{17, 5, 3, -9, -2, -3, 2, 1, -8, -6\}$
(4,7)	$\{17, 5, 3, 2, -2, -3, -9, 1, -8, -6\}$

(5,8)	{17, 5, 3, 2, 1, -3, -9, -2, -8, -6}
(6,8)	{17, 5, 3, 2, 1, -2, -9, -3, -8, -6}
(7,8)	{17, 5, 3, 2, 1, -2, -3, -9, -8, -6}
(8,10)	{17, 5, 3, 2, 1, -2, -3, -6, -8, -9}

You can see from the final row of this list that first the positive entries of the data have been sorted in descending order, and then the negative entries in the data have been sorted in ascending order of absolute value.

Sample Input 2

```
3
9.5
9.4
9.5
```

Sample Output 2

```
2 3
all sorted
```

Note that the input contains real numbers, not just integers. In C++, you should use a floating point type (`float`, `double`, or `long double`) to store these values.

Scoring

The score for each input scenario will be 100% if the correct answer is written to the output file, 20% if the sequence of swaps given sorts the list correctly but is too long, and 0% otherwise. Note that there may be more than one correct output file; it doesn't matter which of these your programme produces.