

## D. Make a Permutation!

time limit per test: 2 seconds  
memory limit per test: 256 megabytes  
input: standard input  
output: standard output

Ivan has an array consisting of  $n$  elements. Each of the elements is an integer from 1 to  $n$ .

Recently Ivan learned about permutations and their lexicographical order. Now he wants to change (replace) *minimum* number of elements in his array in such a way that his array becomes a *permutation* (i.e. each of the integers from 1 to  $n$  was encountered in his array exactly once). If there are multiple ways to do it he wants to find the *lexicographically minimal* permutation among them.

Thus minimizing the number of changes has the first priority, lexicographical minimizing has the second priority.

In order to determine which of the two permutations is lexicographically smaller, we compare their first elements. If they are equal — compare the second, and so on. If we have two permutations  $x$  and  $y$ , then  $x$  is lexicographically smaller if  $x_i < y_i$ , where  $i$  is the first index in which the permutations  $x$  and  $y$  differ.

Determine the array Ivan will obtain after performing all the changes.

### Input

The first line contains an single integer  $n$  ( $2 \leq n \leq 200\,000$ ) — the number of elements in Ivan's array.

The second line contains a sequence of integers  $a_1, a_2, \dots, a_n$  ( $1 \leq a_i \leq n$ ) — the description of Ivan's array.

### Output

In the first line print  $q$  — the minimum number of elements that need to be changed in Ivan's array in order to make his array a permutation. In the second line, print the *lexicographically minimal* permutation which can be obtained from array with  $q$  changes.

### Examples

<b>input</b>
4 3 2 2 3
<b>output</b>
2 1 2 4 3
<b>input</b>
6 4 5 6 3 2 1
<b>output</b>
0 4 5 6 3 2 1
<b>input</b>
10 6 8 4 6 7 1 6 3 4 5
<b>output</b>
3 2 8 4 6 7 1 9 3 10 5

**Note**

In the first example Ivan needs to replace number three in position 1 with number one, and number two in position 3 with number four. Then he will get a permutation  $[1, 2, 4, 3]$  with only two changed numbers — this permutation is lexicographically minimal among all suitable.

In the second example Ivan does not need to change anything because his array already is a permutation.