C. Ceremonial Music (Ceremony)

Problem Statement

After many years of hardwork, you finally won first place in NHSPC! Still feeling excited from the adrenaline rush, you happened to walk past the award ceremony venue just before the ceremony, hearing the sound check for the music on the ceremony.

"This sounds horrible..." You don't think the music is well suited for the ceremony. Fortunately, as a master of algorithms, a musical genius and an IT security guru, you decided to take matters into your own hands, and spice up the ceremonial music.

The ceremonial music is composed of N notes, with its i-th note having pitch a_i. You can modify the music by changing the order of the notes, but in order to not get caught, in one operation, you can only choose an interval with the same starting pitch and ending pitch, then reverse the entire interval.

Formally, in each operation, you choose two indices $1 \le l \le r \le N$ With $a_l = a_r$, and reverse the interval from $a_l, a_{l+1}, \dots, a_r$ to $a_r, a_{r-1}, \dots, a_l$. You can perform this operation arbitrarily many (possibly 0) times.

You think melodies of the form "a low pitch, then a high pitch, then a low pitch" and "a high pitch, then a low pitch, then a high pitch" are bad. So you want to minimize the number of such melodies. In other words, after the operations are done, you want the number of indices $i\,(1 < i < N)$ satisfying $a_{i-1} < a_i > a_{i+1}$ or $a_{i-1} > a_i < a_{i+1}$ to be the lowest possible.

Please write a program that finds the optimal list of notes after performing any number of operations.

Input Format

```
N
a_1  a_2  ...  a_N
```

- N denotes the number of notes in the ceremonial music.
- a_i is the original pitch of the i-th note in the music.

Output Format

```
b_1  b_2  ...  b_N
```

- b_i is the pitch of the i-th note after all of the operations has been done.
- If there's more than one answer with minimal bad melodies, any one of them will be accepted.

Constraints

- $3 \leq N \leq 10^6$
- $1 \leq a_i \leq N$
- All inputs are integers

Example

Sample Input	Sample Output
10 2 2 6 2 1 3 6 4 1 4	2 6 4 1 2 2 6 3 1 4
10 2 5 5 5 10 2 3 3 10 2	2 10 5 5 5 2 3 3 10 2
10 8 4 9 4 6 5 8 2 4 8	8 2 4 9 4 6 5 8 4 8
7 2 3 2 1 4 3 4	2 3 4 3 2 1 4

Scoring

There are 5 subtasks in this problem. The score and additional constraints of each subtask are as follows:

Subtask	Score	Additional constraints
1	9	$N \leq 10$
2	20	$a_i \leq 2$
3	17	$N \leq 1000$
4	21	$N \leq 10^5$
5	33	No other constraints