

E. Marbles

time limit per test: 4 seconds
 memory limit per test: 256 megabytes
 input: standard input
 output: standard output

Monocarp has arranged n colored marbles in a row. The color of the i -th marble is a_i . Monocarp likes ordered things, so he wants to rearrange marbles in such a way that all marbles of the same color form a contiguous segment (and there is only one such segment for each color).

In other words, Monocarp wants to rearrange marbles so that, for every color j , if the leftmost marble of color j is l -th in the row, and the rightmost marble of this color has position r in the row, then every marble from l to r has color j .

To achieve his goal, Monocarp can do the following operation any number of times: choose two neighbouring marbles, and swap them.

You have to calculate the minimum number of operations Monocarp has to perform to rearrange the marbles. Note that the order of segments of marbles having equal color does not matter, it is only required that, for every color, all the marbles of this color form exactly one contiguous segment.

Input

The first line contains one integer n ($2 \leq n \leq 4 \cdot 10^5$) — the number of marbles.

The second line contains an integer sequence a_1, a_2, \dots, a_n ($1 \leq a_i \leq 20$), where a_i is the color of the i -th marble.

Output

Print the minimum number of operations Monocarp has to perform to achieve his goal.

Examples

input	Copy
7 3 4 2 3 4 2 2	
output	Copy
3	

input	Copy
5 20 1 14 10 2	
output	Copy
0	

input	Copy
13 5 5 4 4 3 5 7 6 5 4 4 6 5	
output	Copy
21	

Note

In the first example three operations are enough. Firstly, Monocarp should swap the third and the fourth marbles, so the sequence of colors is $[3, 4, 3, 2, 4, 2, 2]$. Then Monocarp should swap the second and the third marbles, so the sequence is $[3, 3, 4, 2, 4, 2, 2]$. And finally, Monocarp should swap the fourth and the fifth marbles, so the sequence is $[3, 3, 4, 4, 2, 2, 2]$.

Codeforces Round #585 (Div. 2)

Finished

Practice



→ Virtual participation

Virtual contest is a way to take part in past contest, as close as possible to participation on time. It is supported only ICPC mode for virtual contests. If you've seen these problems, a virtual contest is not for you - solve these problems in the archive. If you just want to solve some problem from a contest, a virtual contest is not for you - solve this problem in the archive. Never use someone else's code, read the tutorials or communicate with other person during a virtual contest.

[Start virtual contest](#)

→ Practice

You are registered for practice. You can solve problems unofficially. Results can be found in the contest status and in the bottom of standings.

→ Clone Contest to Mashup

You can clone this contest to a mashup.

[Clone Contest](#)

→ Submit?

Language: GNU G++11 5.1.0

Choose file: [选择文件](#) 未选择任何文件

Be careful: there is 50 points penalty for submission which fails the pretests or resubmission (except failure on the first test, denial of judgement or similar verdicts). "Passed pretests" submission verdict doesn't guarantee that the solution is absolutely correct and it will pass system tests.



[Submit](#)

→ Problem tags

[bitmasks](#) [dp](#)

No tag edit access

→ Contest materials

- Announcement #1 (en) 
- Announcement #2 (ru) 

In the second example there's no need to perform any operations.

• Tutorial (en)



[Codeforces](#) (c) Copyright 2010-2019 Mike Mirzayanov
The only programming contests Web 2.0 platform
Server time: Sep/17/2019 08:52:43^{UTC+8} (f2).
Desktop version, switch to [mobile version](#).
[Privacy Policy](#)

Supported by



ITMO UNIVERSITY