Melody

Input File: `melodyin.txt`
Output File: `melodyout.txt`

Time and Memory Limits: 1 second, 1 GB

Your friend Melody has written a song. Melody's song is a sequence of $\mathbf{N}$ notes, where each note is represented by an integer between $\mathbf{1}$ and $\mathbf{K}$. **$\mathbf{N}$ is always a multiple of three.**

You happen to be very knowledgeable in the trends of modern pop music, and know that a song is *nice* if and only if it consists of the *same sequence of three notes repeated over and over again*, until the end.

For example, the following songs are nice:

- $\mathbf{1, 2, 3, 1, 2, 3, 1, 2, 3}$ (since $\mathbf{1, 2, 3}$ is repeated over and over again)
- $\mathbf{6, 4, 4, 6, 4, 4}$ (since $\mathbf{6, 4, 4}$ is repeated over and over again)
- $\mathbf{8, 8, 8, 8, 8, 8}$ (since $\mathbf{8, 8, 8}$ is repeated over and over again)
- $\mathbf{6, 2, 6}$ (since $\mathbf{6, 2, 6}$ is repeated over and over again).

The following songs are not nice:

- $\mathbf{1, 2, 3, 4, 5, 6}$
- $\mathbf{8, 8, 8, 8, 8, 4}$
- $\mathbf{1, 2, 3, 1, 3, 2, 1, 2, 3}$
- $\mathbf{2, 2, 5, 5, 2, 2}$.

After hearing your advice, Melody would like to change (but not add or remove) some of, or none of, the notes in her song, so that it is nice.

What is the smallest possible number of notes she can change so that her song is nice?

Input

- The first line of input contains the two integers $\mathbf{N}$ and $\mathbf{K}$.
- The next $\mathbf{N}$ lines each contain one integer. The $\mathbf{i}$th of these is $\mathbf{S_i}$, the $\mathbf{i}$th note in Melody's song.

Output

Your program should output a single integer, the smallest possible number of notes Melody can change so that her song is nice.

Sample Input 1

```
6 4
1
2
3
4
3
3
```

Sample Input 2

```
9 10
8
8
3
8
8
3
4
8
3
```

Sample Input 3

```
3 2
2
2
1
```

Sample Output 1

```
2
```

Sample Output 2

```
1
```

Sample Output 3

```
0
```

Explanation

In the first sample input, the song $\mathbf{1, 2, 3, 4, 3, 3}$ could be made into the nice song $\mathbf{1, 2, 3, 1, 2, 3}$. This requires 2 notes (those in the 4th and 5th positions) to be changed, which is the fewest possible.

In the second sample input, the song $\mathbf{8, 8, 3, 8, 8, 3, 4, 8, 3}$ could be made into the nice song $\mathbf{8, 8, 3, 8, 8, 3, 8, 8, 3}$. This requires 1 note (the one in the 7th position) to be changed, which is the fewest possible.

In the third sample input, the song $\mathbf{2, 2, 1}$ is already nice, so no notes need to be changed.

Subtasks & Constraints

For all test cases:

- $\mathbf{3 \le N \le 99\,999}$.
- $\mathbf{N}$ is a multiple of three.
- $\mathbf{1 \le K \le 100\,000}$.
- $\mathbf{1 \le S_i \le K}$, for all $\mathbf{i}$.

Additionally:

- For Subtask 1 (20 points), making Melody's song into the nice song $\mathbf{1, 2, 3, 1, 2, 3}, \ldots$ will require the smallest possible number of changes. See Sample Input 1 for an example.
- For Subtask 2 (30 points), $\mathbf{N \le 30}$ and $\mathbf{K \le 30}$.
- For Subtask 3 (50 points), no special constraints apply.