Back To School '19: A Circular Game

Time limit: 1.0s   **Memory limit:** 64M

At Zeyu's school, a certain circle game has been getting all the craze lately.

The game consists of N identical circular spinners, each containing M equally-sized sectors labelled in order from 1 to M. A small arrow at the top of the spinner points to one of the M sectors to represent the sector it is currently on.

In this game, two players race each other to set all N spinners to point to the same number. They do so by picking a spinner and change the arrow to increase or decrease by 1 of the current sector it is pointing at. Increasing by 1 at the Mth sector will move the arrow to the 1st sector, and decreasing by 1 at the 1st sector will move the arrow to the Mth sector. Doing this will cost them 1 second every time. The winner will be the player who does this the fastest.

Zeyu wants to become the coolest kid in his grade, and aims to do so by becoming the very best at this game. Of course, he isn't very good at it and needs your aid. Help Zeyu find the minimum time to solve a given configuration of circles.

Input Specification

The first line will contain two integers N and M ($1 \leq N \leq 10^5, 1 \leq M \leq 10^9$), the number of spinners and the number of sectors.
The second line will contain N integers ($1 \leq v_i \leq M$), with v_i representing the current sector the ith spinner is pointing at.

Output Specification

In the first and only line of the output, your program should print an integer representing the minimum number of seconds required to set all spinners to the same number.

Constraints

Subtask 1 [5%]

$1 \leq N, M \leq 10^3$

Subtask 2 [15%]

$1 \leq N \leq 10^3$

Subtask 3 [80%]

No additional constraints.

Sample Input 1

```
3 8
2 3 8
```

Sample Output 1

```
3
```

Explanation for Sample 1

By setting all the spinners to 2, we can do it in 3 seconds:

```
3 => 2
8 => 1 => 2
```

This is the fastest way.

Sample Input 2

```
5 10
2 4 3 9 1
```

Sample Output 2

```
7
```

Explanation for Sample 2

By setting all the spinners to 2, we can do it in 7 seconds:

```
4 => 3 => 2
3 => 2
9 => 10 => 1 => 2
1 => 2
```