World

After you won the largest programming contest in the galaxy, competitive programming became very popular in your country. In fact, it became the national sports. But even after such a huge achievement, your mom still doesn't accept you as the best. So, in order to prove yourself to her, you decided to solve the hardest problem in the millennium! The problem is as follows:

You are given an array size n containing m-dimensional points. We define the distance $D(A, B)$ between two points $A(x_1, x_2, \ldots, x_m)$ and $B(y_1, y_2, \ldots, y_m)$ as:

$$D(A, B) = \sum_{i=1}^{m} |x_i - y_i|$$

For example, consider the points $A(4, 5, 6)$ and $B(0, 5, 9)$. Then, $D(A, B) = |4 - 0| + |5 - 5| + |6 - 9| = 7$

Your task is to divide the array into some subarrays such that each point belongs to exactly one subarray. A subarray is a contiguous portion of an array, formed by removing some elements (possibly zero) from its beginning and end. The cost of a subarray is the maximum distance between two points in the subarray. You want to to maximize the sum of costs of the subarrays.

What is the maximum sum of costs over all possible divisions?

Input

Read the input from the standard input in the following format:

- line 1: $n \quad m$
- line $1 + i$ $(1 \leq i \leq n)$: $x[i][1] \;\; x[i][2] \;\; \ldots \;\; x[i][m]$

Here $x[i][j]$ is the j-th component of i-th point.

Output

Write the output to the standard output in the following format:

- line 1: The maximum sum of costs over all possible divisions.

Constraints

- $1 \leq n \leq 100\,000$

- $1 \leq m \leq 10$
- $0 \leq x[i][j] \leq 10^9$ (for all $1 \leq i \leq n$ and $1 \leq j \leq m$)

Subtasks

1. (5 points) $n \leq 80$
2. (8 points) $n \leq 3000$, $m = 1$
3. (5 points) $m = 1$, $0 \leq x[i][1] \leq 1$ (for all $1 \leq i \leq n$)
4. (10 points) $0 \leq x[i][j] \leq 1$ (for all $1 \leq i \leq n$ and $1 \leq j \leq m$)
5. (21 points) $m = 1$
6. (12 points) $n \leq 500$
7. (8 points) $n \leq 3000$
8. (31 points) No further constraints.

Example 1

```
7 1
1
3
0
1
7
100
5
```

The correct output is:

```
104
```

In this case the points are one-dimensional and the array is $[1, 3, 0, 1, 7, 100, 5]$. The optimal way to divide the array would be $[1, 3, 0]$, $[1, 7]$ and $[100, 5]$ with costs $3, 6$ and 95 respectively.

Example 2

```
4 2
0 5
5 5
10 10
5 5
```

The correct output is:

```
15
```

In this case the points are two-dimensional and the array is $[(0, 5), (5, 5), (10, 10), (5, 5)]$. The optimal way to divide the array would be $[(0, 5), (5, 5), (10, 10)]$, and $[(5, 5)]$, with costs 15 and 0. respectively.

Example 3

```
10 3
5 10 9
1 2 3
9 8 1
5 4 9
10 15 9
0 10 11
14 15 18
9 1 2
15 10 9
0 0 0
```

The correct output is:

```
120
```