E. Pastoral Oddities

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

In the land of Bovinia there are n pastures, but no paths connecting the pastures. Of course, this is a terrible situation, so Kevin Sun is planning to rectify it by constructing m undirected paths connecting pairs of distinct pastures. To make transportation more efficient, he also plans to pave some of these new paths.

Kevin is very particular about certain aspects of path-paving. Since he loves odd numbers, he wants each pasture to have an odd number of paved paths connected to it. Thus we call a paving sunny if each pasture is incident to an odd number of paved paths. He also enjoys short paths more than long paths, so he would like the longest paved path to be as short as possible. After adding each path, Kevin wants to know if a sunny paving exists for the paths of Bovinia, and if at least one does, the minimum possible length of the longest path in such a paving. Note that "longest path" here means maximum-weight edge.

Input

The first line contains two integers n ($2 \le n \le 100\,000$) and m ($1 \le m \le 300\,000$), denoting the number of pastures and paths, respectively. The next m lines each contain three integers a_i, b_i and l_i, describing the i-th path. The i-th path connects pastures a_i and b_i ($1 \le a_i, b_i \le n$; $a_i \ne b_i$) and has length l_i ($1 \le l_i \le 10^9$). Paths are given in the order in which they are constructed.

Output

Output m lines. The i-th line should contain a single integer denoting the minimum possible length of the longest path (maximum-weight edge) in a sunny paving using only the first i paths. If Kevin cannot pave a set of paths so that each pasture is incident to an odd number of paved paths, output -1.

Note that the paving is only hypothetical—your answer after adding the i-th path should not be affected by any of your previous answers.

Examples

input

```
4 4
1 3 4
2 4 8
1 2 2
3 4 3
```

output

```
-1
8
8
3
```

input

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

output

```
-1
-1
```

input

```
4 10
2 1 987
3 2 829
4 1 768
4 2 608
3 4 593
3 2 488
4 2 334
2 1 204
1 3 114
1 4 39
```

output

```
-1
-1
829
829
768
768
768
488
334
204
```

Note

For the first sample, these are the paths that Kevin should pave after building the i-th path:

1. No set of paths works.
2. Paths 1 (length 4) and 2 (length 8).
3. Paths 1 (length 4) and 2 (length 8).
4. Paths 3 (length 2) and 4 (length 3).

In the second sample, there never exists a paving that makes Kevin happy.