Restore Array

Your task is to determine one possible **binary** array A of length N that abides by M given constraints of the form:
$(l, r, k, value)$ - the k-th smallest element in subarray $A[l..r]$ is $value$ ($0 \leq l \leq r < N$, $1 \leq k \leq r - l + 1$, $0 \leq value \leq 1$). Please note that **array A is 0-indexed**.

Input

The first line of input contains two integers N and M ($1 \leq N \leq 5\,000$, $1 \leq M \leq 10\,000$) - the length of array A and the number of constraints.

The next M lines describe the constraints. Each line contains four integers l_i, r_i, k_i, $value_i$, describing the i-th constraint.

Output

The first line of the output contains N integers - one possible **binary** array A. If there are several that abide by all M constraints you may output any of them. If there is no such array you must instead output the single integer -1.

Subtasks

(1) $1 \leq N \leq 18$, $1 \leq M \leq 200$ **(7 points)**
(2) $1 \leq N \leq 5\,000$, $1 \leq M \leq 10\,000$, for all constraints $k = 1$ holds **(13 points)**
(3) $1 \leq N \leq 5\,000$, $1 \leq M \leq 10\,000$, for all constraints $k = 1$ or $k = (r - l + 1)$ holds **(25 points)**
(4) $1 \leq N \leq 5000$, $1 \leq M \leq 10\,000$ **(55 points)**

Example(s):

Standard Input	Standard Output
4 5 0 1 2 1 0 2 2 0 2 2 1 0 0 1 1 0 1 2 1 0	0 1 0 0

Explanation:

There are several binary arrays that abide by all the constraints. One of them is `0 1 0 0` because:

(1) The 2-nd smallest element among **`0 1`** ~~`0 0`~~ is `1`.

(2) The 2-nd smallest element among **`0 1 0`** ~~`0`~~ is `0`.

(3) The 1-st smallest element among ~~`0 1`~~ **`0`** ~~`0`~~ is `0`.

(4) The 1-st smallest element among **`0 1`** ~~`0 0`~~ is `0`.

(5) The 1-st smallest element among ~~`0`~~ **`1 0`** ~~`0`~~ is `0`.