A. Alyona and mex

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Alyona's mother wants to present an array of n non-negative integers to Alyona. The array should be special.

Alyona is a capricious girl so after she gets the array, she inspects m of its subarrays. Subarray is a set of some subsequent elements of the array. The i-th subarray is described with two integers l_i and r_i, and its elements are $a[l_i], a[l_i + 1], ..., a[r_i]$.

Alyona is going to find *mex* for each of the chosen subarrays. Among these m *mexes* the girl is going to find the smallest. She wants this minimum *mex* to be as large as possible.

You are to find an array a of n elements so that the minimum *mex* among those chosen by Alyona subarrays is as large as possible.

The *mex* of a set S is a minimum possible non-negative integer that is not in S.

Input

The first line contains two integers n and m $(1 \le n, m \le 10^5)$.

The next m lines contain information about the subarrays chosen by Alyona. The i-th of these lines contains two integers l_i and r_i $(1 \le l_i \le r_i \le n)$, that describe the subarray $a[l_i], a[l_i + 1], ..., a[r_i]$.

Output

In the first line print single integer — the maximum possible minimum *mex*.

In the second line print n integers — the array a. All the elements in a should be between 0 and 10^9.

It is guaranteed that there is an optimal answer in which all the elements in a are between 0 and 10^9.

If there are multiple solutions, print any of them.

Examples

input
5 3 1 3 2 5 4 5
output
2 1 0 2 1 0

input
4 2 1 4 2 4
output
3 5 2 0 1

Note

The first example: the *mex* of the subarray $(1, 3)$ is equal to 3, the *mex* of the subarray $(2, 5)$ is equal to 3, the *mex* of the subarray $(4, 5)$ is equal to 2 as well, thus the minumal *mex* among the subarrays chosen by Alyona is equal to 2.