

Problem I. Interesting Coloring

Input file: *standard input*
Output file: *standard output*
Time limit: 2 seconds
Memory limit: 512 mebibytes

Given is an undirected simple connected graph, consisting of N vertices and M edges.

The vertices of this graph are enumerated by sequential integers from 1 to N , and the edges are enumerated by sequential integers from 1 to M , respectively. Edge i connects vertex u_i and vertex v_i .

The following special property holds for this graph: for every edge i ($1 \leq i \leq M$), there exists a path connecting u_i and v_i that does not contain this edge. We will call such path a *bypass path* of edge i .

There may be more than one bypass path for the same edge.

We will color the edges by colors enumerated by sequential integers from 1 to M , assigning exactly one color to every edge. Some colors may be left unused, others may be used more than once.

The coloring of the edges is called *interesting* if the following properties hold:

- If two edges have a common vertex, their colors are different.
- For every edge, there exists a *special* bypass path: a bypass path containing the edges colored with no more than 8 different colors.

Your task is to find an interesting coloring and, for each of the M edges, print any set of colors that can be used to build a special bypass path for that edge.

It can be shown that, under the constraints above, there exists at least one interesting coloring.

Input

The first line of input contains two integers N and M ($3 \leq N \leq 5555$, $3 \leq M \leq \min(N(N-1)/2, 9999)$).

The i -th of the M following lines describes the i -th edge and contains two integers u_i and v_i ($1 \leq u_i < v_i \leq N$).

You may assume that each pair (u, v) appears in the list at most once, that the given graph is connected and that, after removal of any edge (u, v) , there still exists a bypass path connecting u and v .

Output

Print any interesting coloring in the following format.

On the first line, print M integers. The i -th of these integers, C_i , must be the color of the i -th edge ($1 \leq C_i \leq M$).

Then print M lines. The i -th of these lines describes the color set of the special bypass path for edge i . This line must start with the integer k_i ($1 \leq k_i \leq 8$): the number of the colors in the list. It must be followed by k_i pairwise distinct integers between 1 and M : the list of colors. The colors can be printed in any order. There must exist a special bypass path between u_i and v_i which does not use any colors except the colors in the list. Note that this means the list of colors does not have to be the minimal possible, and there can even be a path that uses only a part of the list: the checking program only makes sure that the listed colors are sufficient.

Example

standard input	standard output
10 11	1 2 3 4 5 6 7 8 9 10 5
1 2	3 2 3 5
2 3	3 1 3 5
3 4	3 1 2 5
4 5	6 5 6 7 8 9 10
5 6	7 4 5 6 7 8 9 10
6 7	6 4 5 7 8 9 10
7 8	6 4 5 6 8 9 10
8 9	6 4 5 6 7 9 10
9 10	6 4 5 6 7 8 10
1 10	8 4 5 6 7 8 9 1 2
1 4	3 1 2 3

Note

In the example, there are two bypass paths for the first edge.

The longer one contains 9 colors (from 2 to 10), so it is not special.

The shorter one consists of the edges 2, 3, and 11 (colors 2, 3, and 5), so it is special.