Colorful Cubes

Bill Games, an excellent programmer, spent Easter with his grandparents. In their old house he came across wooden cubes - a child's toy. When he was a child, he used to build castles and towers and pyramids of these colorful cubes. He started to play with them again. However, the problem which he tries to solve today is much more complicated than building a simple pyramid.

Each face of a cube is colored with one color. Bill wants to build a tower from all cubes he has. This means to stack all the cubes in one column, one on another. Bill does not want to put the cubes in arbitrary order - the bottom face of every cube (except the bottom cube which is lying on the floor) should have the same color as the top face of the cube below it.

Input file specification

The first line of the input file consists of two numbers M(1<=M<=100) and N(1<=N<=500). M is the number of colors used (colors are numbered 1...M) and N is the number of cubes (cubes are numbered 1...N in the order as they appear on the input).

Next N lines represent cubes 1,2,...,N in this order. A cube is described by six numbers giving colors of its faces in the following order: front, back, right, left, bottom, and top face.

Output file specification

Given the cubes described in the input file determine how to arrange them into a tower. Every cube has to be used exactly once. You need to find only one solution. Assume that the solution exists.

The output file consists of N lines. The i-th line contains the description of the cube on the i-th position in the tower, counting from bottom. The description of a cube consists of seven numbers. The first number is the number of the cube (the order of the cube in the input file) and the following six numbers represent colors of the faces in the following order: front, back, right, left, bottom, and top face. Notice that cubes can be rotated.

Example

Input file #1:
6 2
1 2 3 4 5 6
2 1 3 4 5 6

Output file #1:
1 6 5 3 4 1 2
2 6 5 3 4 2 1

Input file #2:
3 3
1 2 2 2 1 2
3 3 3 3 3 3
3 2 1 1 1 1

Output file #2:

1 1 2 2 2 1 2
3 1 1 1 1 2 3
2 3 3 3 3 3 3