When a competitive programmer thinks of a tree, it's not the same tree that a regural person thinks of. Luckily, in this task, both meanings are correct.

Nikola loves nature and often walks in the woods, looking at the trees and admiring their size, node degrees, paradoxically regular randomness, etc. In this day of lovely spring, there are even more reasons to look above into these magnificent creatures: the trees are full of color and that captured Nikola's attention.

So one day he observed a large tree with N nodes, seeing a color in each node. Was there a flower, an animal, or Nikola simply went mad, it's hard to say. But he was looking at that tree for a very long time, and in an N × N matrix he wrote, for each two nodes, the number of distinct colors on a unique simple path between (and including) these two nodes. Sadly, by admiring all those colors, Nikola completely forgot what the tree looked like and which colors were in the nodes.

So he needs your help. From the matrix he wrote, determine a possible tree and possible colors of the respective nodes. Colors should be denoted by numbers from {1, 2, …, N}. It is guaranteed that Nikola did not make a mistake; in other words, a solution will exist.

INPUT

The first line contains a subtask number (1, 2 or 3) from the *Scoring* section below.

The second line contains an integer N (1 ≤ N ≤ 3000), the number of nodes in the tree, which are denoted 1, 2, ..., N.

Each of the next N lines contains N integers, where *j*-th number in *i*-th line equals the number of distinct colors on the path from node *i* to node *j*.

OUTPUT

The first line should contain N space-separated integers between 1 and N: colors of the nodes 1, 2, ..., N.

Each of the next N – 1 lines should contain two integers A, B representing the edge (neighboring nodes) in the tree. Order of the edges and nodes inside an edge is arbitrary.

SCORING

subtask	points	additional constraints
1	18	all numbers in the matrix equal 1 or 2
2	25	there is a solution where the tree is a chain of nodes 1, 2, ..., N in that order, i.e., the edges are (*i*, *i* + 1) for each *i* = 1, ..., N – 1
3	57	no additional constraints

(See examples on the next page.)

SAMPLE TESTS

input

```
3
5
1 2 2 2 3
2 1 3 3 2
2 3 1 3 4
2 3 3 1 3
3 2 4 3 1
```

output

```
1 2 3 4 4
1 2
1 3
1 4
2 5
```

input

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

output

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

input

```
1
5
1 2 2 2 2
2 1 1 2 2
2 1 1 2 2
2 2 2 1 2
2 2 2 2 1
```

output

```
1 2 2 1 2
1 2
2 3
2 4
1 5
```