A company decides to simulate on computer the process of manufacturing its own goods. In order to do that, it makes the following observations:

1. The whole process can be split into several steps, between them there are some dependencies. This can be represented by a diagram (graph), which we suppose to be only one for all goods produced by company as in figure 1;

2. First step designates the start of manufacturing process; there is only one first step, denoted by the number 1;

3. There are not steps isolated or outside the process (every step is linked by a path with the first step);

4. Some steps are total dependants; so, we claim that the step \( i \) is total dependant of step \( j \) if every path in the fabrication process cannot arrive to \( i \) without passing through \( j \).

So, all steps are total dependants of step 1.

Example: In the process shown by the figure 1 the step 4 is total dependant of step 3, steps 5, 6 and 7 are total dependants of 4 (hearse of 3), but step 3 is not total dependant of step 2.

The Computing Center Dept. of company notes that whole manufacturing process is easier to be controlled if it would be structured by a tree, as follows:

- All steps of manufacturing process are nodes of the tree;
- Each node ensures total dependence of all its own descendants;

The tree associated to the diagram from figure 1 is shown in figure 2.

Your task is to write a program that builds this dependence tree.

Input
The input file contains several input data sets. An input data set has the following format:

\[ n \]
\[ a_{11} \, a_{12} \, \ldots \, a_{1n} \]
\[ a_{21} \, a_{22} \, \ldots \, a_{2n} \]
\[ \ldots \]
\[ a_{n1} \, a_{n2} \, \ldots \, a_{nn} \]

where \( a_{ij} = 1 \) if step \( j \) follows directly step \( i \) in the process diagram, otherwise \( a_{ij} = 0 \).

Output
At output, the program must write \( n - 1 \) lines for every input data set; each line has the format:

\[ i \, j \]

with the meaning that node \( j \) is a direct descendant of node \( i \) in the tree. The pair \((i, j)\) follows \((i', j')\) if and only if \( i' < i \) or (\( i'_1 = i_1 \) and \( j'_1 < j_1 \)).

Sample Input

\[ 10 \]
\[ 0 \, 1 \, 0 \, 0 \, 0 \, 0 \, 0 \, 0 \, 0 \, 0 \]
\[ 0 \, 0 \, 1 \, 0 \, 0 \, 0 \, 0 \, 0 \, 0 \, 0 \]
\[ 0 \, 0 \, 0 \, 1 \, 0 \, 0 \, 0 \, 0 \, 0 \, 0 \]
\[ 0 \, 0 \, 0 \, 0 \, 1 \, 0 \, 0 \, 0 \, 0 \, 0 \]
\[ 0 \, 0 \, 0 \, 0 \, 0 \, 1 \, 0 \, 0 \, 0 \, 0 \]
\[ 0 \, 0 \, 0 \, 0 \, 0 \, 0 \, 1 \, 0 \, 0 \, 0 \]
\[ 0 \, 0 \, 0 \, 0 \, 0 \, 0 \, 0 \, 1 \, 0 \, 0 \]
\[ 0 \, 0 \, 0 \, 0 \, 0 \, 0 \, 0 \, 0 \, 1 \, 0 \]
\[ 0 \, 0 \, 0 \, 0 \, 0 \, 0 \, 0 \, 0 \, 0 \, 1 \]

Sample Output

\[ 1 \, 2 \]
\[ 1 \, 3 \]
\[ 3 \, 4 \]
\[ 3 \, 5 \]
\[ 4 \, 6 \]
\[ 4 \, 7 \]
\[ 7 \, 8 \]
\[ 8 \, 9 \]
\[ 8 \, 10 \]