2954　Task Sequences

Tom has received a lot of tasks from his boss, which are boring to deal with by hand. Fortunately, Tom got a special machine - Advanced Computing Machine (ACM) to help him.

ACM works in a really special way. The machine can finish one task in a short time, after it's finishing one task, it should smoothly move to the next one, otherwise the machine will stop automatically. You must start it up again to make it continue working. Of course, the machine cannot move arbitrarily from one task to another. So each time before it starts up, one task sequence should be well scheduled. Specially, a single task also can be regarded as a sequence. In the sequence, the machine should be able to smoothly move from one task to its successor (if exists). After started up, the machine always works according to the task sequence, and stops automatically when it finishes the last one. If not all the tasks have been finished, the machine has to start up again and works according to a new sequence. Of course, the finished tasks can't be scheduled again.

For some unknown reasons, it was guaranteed that for any two tasks i and j, the machine can smoothly move from i to j or from j to i or both. Because the startup process is quite slow, Tom would like to schedule the task sequences properly, so that all the tasks can be completed with minimal number of startup times. It is your task to help him achieve this goal.

Input

Input contains several testcases. For each testcase, the first line contains only one integer n, $(0 < n \le 1,000)$, representing the number of tasks Tom has received. Then n lines follow. Each line contains n integers, '0' or '1', separated by white spaces. If the j-th integer in the i-th line is 1, then the machine can smoothly move from task i to task j, otherwise the machine can not smoothly move from task i to task j. The tasks are numbered from 1 to n.

Input is terminated by end of file.

Output

For each testcase, the first line of output is only one integer k, the minimal number of startup times needed. And $2k$ lines follow, to describe the k task sequences. For each task sequence, the first line should contain one integer m, representing the number of tasks in the sequence. And the second line should contain m integers, representing the order of the m tasks in the sequence. Two consecutive integers in the same line should be separated by just one white space. Extra spaces are not allowed. There may be several solutions, any appropriate one is accepted.

Sample Input

```
3
0 1 1
1 0 1
0 0 0
```

Sample Output

```
1
3
2 1 3
```