

D2. Hyperspace Jump (hard)

time limit per test: 3 seconds

memory limit per test: 256 megabytes

input: standard input

output: standard output

It is now 125 years later, but humanity is still on the run from a humanoid-cyborg race determined to destroy it. Or perhaps we are getting some stories mixed up here... In any case, the fleet is now smaller. However, in a recent upgrade, all the navigation systems have been outfitted with higher-dimensional, linear-algebraic jump processors.

Now, in order to make a jump, a ship's captain needs to specify a *subspace* of the d -dimensional space in which the events are taking place. She does so by providing a generating set of vectors for that subspace.

Princess Heidi has received such a set from the captain of each of m ships. Again, she would like to group up those ships whose hyperspace jump subspaces are equal. To do so, she wants to assign a group number between 1 and m to each of the ships, so that two ships have the same group number if and only if their corresponding subspaces are equal (even though they might be given using different sets of vectors).

Help Heidi!

Input

The first line of the input contains two space-separated integers m and d ($2 \leq m \leq 30\,000$, $1 \leq d \leq 5$) – the number of ships and the dimension of the full underlying vector space, respectively. Next, the m subspaces are described, one after another. The i -th subspace, which corresponds to the i -th ship, is described as follows:

The first line contains one integer k_i ($1 \leq k_i \leq d$). Then k_i lines follow, the j -th of them describing the j -th vector sent by the i -th ship. Each of the j lines consists of d space-separated integers $a_j, j = 1, \dots, d$, that describe the vector ; it holds that $|a_j| \leq 250$. The i -th subspace is the linear span of these k_i vectors.

Output

Output m space-separated integers g_1, \dots, g_m , where g_i denotes the group number assigned to the i -th ship. That is, for any $1 \leq i < j \leq m$, the following should hold: $g_i = g_j$ if and only if the i -th and the j -th subspaces are equal. In addition, the sequence (g_1, g_2, \dots, g_m) should be lexicographically minimal among all sequences with that property.

Example

input

```
8 2
1
5 0
1
0 1
1
0 1
2
0 6
0 1
2
0 1
1 0
2
-5 -5
4 3
2
1 1
0 1
2
1 0
1 0
```

output

1 2 2 2 3 3 3 1

Note

In the sample testcase, the first and the last subspace are equal, subspaces 2 to 4 are equal, and subspaces 5 to 7 are equal.

Recall that two subspaces, one given as the span of vectors and another given as the span of vectors , are equal if each vector v_i can be written as a linear combination of vectors w_1, \dots, w_k (that is, there exist coefficients such that $v_i = \alpha_1 w_1 + \dots + \alpha_k w_k$) and, similarly, each vector w_i can be written as a linear combination of vectors v_1, \dots, v_n .

Recall that a sequence (g_1, g_2, \dots, g_m) is lexicographically smaller than a sequence (h_1, h_2, \dots, h_m) if there exists an index i , $1 \leq i \leq m$, such that $g_i < h_i$ and $g_j = h_j$ for all $j < i$.