

Submit: `happy.c` / `happy.cpp` / `happy.pas`
 Input: `svio`
 Output: `svio`

Time limit: seconds
 Memory limit: 64 MB
 Points: 100

The general elections in Algotopolis are knocking on the door and you would like to win them again (don't ask us why). Recently, you have spent a great amount of public money on researching what makes your citizens feel happy, which is of course going to be very important now. To be more accurate, you know what exactly every man and woman in your country loves. How? Everyone belongs to some⁰ clubs. For example, if somebody belongs to a computer club and an animal club, you can increase his happiness by buying a new elephant or a new mouse. However, if you make somebody too happy, he becomes mad and his happiness falls to zero.

To be more exact, happiness of each citizen can be characterized by a single number n_i ($0 \leq n_i \leq 6$). If you do something, which is good for this particular citizen, n_i increases by 1, but if it already was at the maximum, it drops to 0. In other words, $n'_i = (n_i + 1) \bmod 7$.

You know the initial amount n_i of happiness of every citizen and for each possible action ($1 \leq j \leq K$) you also know a set S_j of citizens who will be made happier. You would like to make all citizens absolutely happy ($n_i = 6$ for every i) or to discover that it's impossible to achieve.

Since the final happiness of citizens doesn't depend on the order of the actions you choose, it suffices describe your choice by a sequence of integers a_1, \dots, a_K such that the j -th action will be performed exactly a_j times.

Formally speaking, you should find a_1, \dots, a_K satisfying:

$$\forall i : \left(n_i + \sum_{j: i \in S_j} a_j \right) \bmod 7 = 6.$$

Input

The input starts with two integers: N and K , where N ($1 \leq N \leq 400$) is the number of citizens* and K ($1 \leq K \leq 400$) is the number of possible actions.[†] Then it contains N integers n_1, \dots, n_N ($0 \leq n_i \leq 6$): n_i stands for the initial happiness of the i -th citizen.

Finally, the input contains the description of the clubs. There are K blocks S_1, \dots, S_K . The j -th block contains a sequence of integers terminated by -1 , listing the citizens in the j -th club, and therefore made happier by the j -th action.

Output

The output should contain a single line containing the integers a_1, \dots, a_K , where a_j ($0 \leq a_j \leq 1\,000\,000$) tells how many times you would like to perform the j -th action. If there are multiple solutions, you can print an arbitrary one. If there is no solution, output a string 'I can't win.'

Example 1

Input:
 3 4
 4
 4
 4
 1 2 3 -1
 2 3 -1
 1 2 -1
 3 1 -1

Output:
 2 0 0 0

Another possible output:
 0 1 1 1

Example 2

Input:
 2 1
 2
 1
 1 2 -1

Output:
 I can't win.

⁰ possibly 0

* It's a small world, isn't it?

[†] ... each corresponding to one club.