

Problem D. Handling a Spaceship

Input file: *standard input*
Output file: *standard output*
Time limit: 1 second
Memory limit: 256 mebibytes

This is an interactive problem.

Wedge Antilles has infiltrated a new secret imperial spaceship which uses N -dimensional hyperspace to move around the Galaxy faster than any other spaceship. Fortunately, there is nobody onboard now, but the spaceship is heading fast in some direction!

The spaceship uses N selectors to choose speed, each of them can be set in one of the M positions. The i -th selector corresponds to i -th *direction* X_i , where X_i is a N -dimensional vector; when it is set to j -th position, the vector $K_{ij} \cdot X_i$ is added to the total speed of the spaceship. Here, each K_{ij} is an integer known as *transmission coefficient* of j -th gear on i -th selector. Therefore, the total speed of the spaceship is an N -dimensional vector defined as $S = \sum_1^N K_{ig_i} X_i$, where g_i is the position the i -th selector is set to.

Wedge knows that, in order to make spaceship driving convenient, the *directions* and *transmission coefficients* of any spaceship, including this one, satisfy some additional constraints. First of all, the *directions* are chosen in such a way that for any desired N -dimensional speed vector S , it's possible to choose *coefficients* c_i such that $S = \sum_1^N c_i X_i$ (note that c_i may or may not be equal to any of the K_{ij}). Additionally, K_{ij} is always less than K_{it} if $j < t$. And the last but not least, for each selector, one of the *transmission coefficients* is equal to zero.

As we've already mentioned, the spaceship is now moving in some direction through the hyperspace, and Wedge doesn't know the *directions* and the *transmission coefficients*. He wants to stop the spaceship. In order to do that, he needs to find the positions for each selector such that the total speed S will be equal to zero. All he can do now is to set each selector to some position (that is, choose g_i for each i) and compute the resulting speed.

Your program has to play Wedge's role and choose positions of each selector. The jury's program will tell your program the resulting speed. Wedge is limited in time, so you could make no more than 120 queries.

Interaction Protocol

At first, your program will be given two integers N and M ($1 \leq N, M \leq 100$). After that, your program has to determine positions of each selector that will produce zero speed.

Though in real world the spaceship is very fast, its hyperspace speed is very limited. Wedge knows that all *transmission coefficients* and components of *directions* don't exceed 1000 by absolute value.

To make a query, the program must print the question mark ("?"), a space, and then N integers g_1, g_2, \dots, g_N separated by spaces. Each number must be in the range from 1 to M . This will mean that Wedge sets the first selector to position g_1 , the second selector to position g_2 , and so on. Then your program must read N integers: the coordinates of the resulting speed vector S .

When your program is ready to print the answer, it must print the English letter "A", a space, and then N integers g_1, g_2, \dots, g_N separated by spaces. After that, your program must exit

immediately.

Don't forget to flush the output after each query.

Please note: if for some query, your program receives the number 987 654 321 as the first integer of the answer, it means that your solution is wrong, and your program must exit immediately. In that case, you can hope to receive the proper outcome (“Wrong Answer”, “Presentation Error”, etc.). If your program does not exit in such situation, you will most likely receive a random outcome instead of what actually happened.

Examples

standard input	standard output
2 2 0 -1 2 0 0 0	? 1 1 ? 2 2 ? 1 2 A 1 2
2 3 0 -2 200 0 0 -1	? 1 1 ? 3 3 ? 1 2 A 1 3

Note

In the first example, there are two selectors and two positions for each of them. The *directions* are the following: $X_1 = (1, 0)$, $X_2 = (0, 1)$. The *transmission coefficients* for the first selector are $K_{11} = 0$, $K_{12} = 2$, and for the second selector, they are $K_{21} = -1$, $K_{22} = 0$. Therefore, to achieve zero speed, Wedge needs to set the first selector in the first position, and the second selector in the second position.

In the second example, there are also two selectors, but now, each of them has three different positions. The *directions* and *transmission coefficients* are the following:

$$\begin{aligned} X_1 &= (1, 0), & X_2 &= (0, 1); \\ K_{11} &= 0, & K_{12} &= 100, & K_{13} &= 200; \\ K_{21} &= -2, & K_{22} &= -1, & K_{23} &= 0. \end{aligned}$$