Problem
Guess the Permutation
Time limit: 3 seconds

This is an interactive problem. Remember to flush the output buffer after every print. To flush your output, you can use:

- `fflush(stdout)` or `cout.flush()` in C/C++;
- `System.out.flush()` in Java and Kotlin;
- `sys.stdout.flush()` in Python;
- See the documentation for other languages.

Little Cyan Fish has a secret permutation $p_0, p_1, \ldots, p_n$ of length $n + 1$, consisting of integers from 0 to n. He wants you to guess the secret permutation. To do that, Little Cyan Fish allows you to perform no more than $1\,024$ queries in the following form:

- `?` i j: Get the index k ($0 \le k \le n$) such that $p_k = |p_i - p_j|$.

Of course, to perform a valid query, you must guarantee that i and j are two integers within $[0, n]$. For example, if $p_0 = 2, p_1 = 0, p_2 = 1$, then the query "`? 0 2`" will give you the index 2 since $|p_0 - p_2| = |2 - 1| = 1 = p_2$.

Little Cyan Fish wants you to come up with a strategy to find the permutation. Of course, Little Cyan Fish does not want you to win the game too easily. Therefore, he will **adaptively** change the permutation p to trap you (see the "Interaction Protocol" for more details).

Input

There are multiple test cases. The first line of the input contains a single integer T ($1 \le T \le 100$), indicating the number of test cases.

For each test case, the first line of the input contains a single integer n ($1 \le n \le 1\,000$). Then, the interaction begins.

Interaction Protocol

To make a query, print a line in the following format:

`?` i j

where i and j are two integers from 0 to n. You must ensure that the number of queries you make does not exceed $1\,024$. As a response to the query, you will get an integer k such that $p_k = |p_i - p_j|$.

To report the answer, output a line in the following format:

`!` p_0 p_1 $\ldots$ p_n

where p is the permutation you wish to report.

After outputting a query or reporting the answer, do not forget to output a newline character and flush the output stream.

In this problem, the interactor is **adaptive**. This means that the permutation p might change depending on your queries, as long as it does not contradict previous queries you asked.

Read	Sample Interaction 1	Write
1 3		
	? 2 2	
1		
	? 0 3	
3		
	! 2 0 3 1	

Testing Tool: A testing tool is provided to help contestants develop and test their solutions. You can download this tool from the attachments. Executing the tool with a "`-h`" option should describe how to use the tool. The testing tool will only implement some test scenarios and only some functionality of the real judge program.