

Yet Another Contest 2 P2 - Secret Sequence

Time limit: 3.0s **Memory limit:** 256M
Java: 10.0s

The combination lock to Josh's locker requires a sequence of N specific integers to be input in a specific order. Josh bets that you won't be able to find this sequence!

Josh allows you to make up to N different queries in order to determine the sequence. In each query, you can select two integers l, r such that $1 \leq l < r \leq N$. Then, he will tell you the **bitwise XOR** of all integers from l -th to the r -th (1-indexed) position in the sequence.

Let M be the minimum value of $r - l$ amongst all queries. To make things harder, Josh challenges you to maximise the value of M whilst still finding the sequence!

Can you successfully find Josh's sequence?

Constraints

$$3 \leq N \leq 10^5$$

All integers in Josh's sequence are greater than 0 and strictly less than 2^{31} .

If you find Josh's sequence correctly, with the maximum possible value of M , you will score 100% of the points.

If you find Josh's sequence correctly, but do not maximise the value of M , you will score 30% of the points.

Otherwise, you will score 0% of the points.

Interaction

This is an interactive problem, where **the interactor is adaptive**.

At first, you should read in a single integer on a single line, representing the value of N .

Then, you will have the opportunity to make up to N queries. To make a query, output a line in the form `? l r`. Then, you should read in a single integer on a single line, representing the bitwise XOR of all elements from the l -th to the r -th positions in the sequence.

Finally, once you have made all of your queries, you should output a line in the form `! a_1, a_2, ..., a_N`, where a is the sequence which you believe is Josh's sequence.

If at any point you format your queries incorrectly, or ask more than N queries, the interactor will respond with -1 on a single line. After receiving this feedback, you should terminate your program to receive a Wrong Answer verdict; otherwise, your program will receive an arbitrary verdict.

If you successfully guess the correct sequence, then you will receive an Accepted verdict for that test case. Your final score for the problem will be equal to the minimum percentage of points received for any test case.

Please note that you may need to flush `stdout` after each operation, or interaction may halt. In C++, this can be done with `fflush(stdout)` or `cout << flush` (depending on whether you use `printf` or `cout`). In Java, this can be done with `System.out.flush()`. In Python, you can use `sys.stdout.flush()`.

Sample Interaction

`>>>` denotes your output. Do not print this out.

```
3
>>> ? 1 2
5
>> ? 1 3
0
>> ? 2 3
1
>> ! 1 4 5
```

Explanation

Josh's sequence is the sequence $[1, 4, 5]$.

For the first query, the bitwise XOR of the elements between the 1-st and 2-nd positions is $1 \oplus 4 = 5$.

For the second query, the bitwise XOR of the elements between the 1-st and 3-rd positions is $1 \oplus 4 \oplus 5 = 0$.

For the third query, the bitwise XOR of the elements between the 2-nd and 3-rd positions is $4 \oplus 5 = 1$.

Here, \oplus denotes the bitwise XOR operator.

The sequence has been correctly guessed, with $M = \min(2 - 1, 3 - 1, 3 - 2) = 1$. It can be shown that this is the maximum possible M such that the sequence is guessable in $N = 3$ queries, so this would score 100% of the points.