Permutation

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	1024 megabytes

The original version of this problem asks you to find the position of the second largest number in an interval for a lot of times, but none of the problem setters knows how to solve that problem, so...

— Dummy Animal Crew

This is an interactive problem.

There's an unknown permutation of length n. You want to determine the position of number n in this permutation.

To do that, you can ask the following question:

- Choose an interval $[l, r]$ ($l < r$) and ask for the **position** of the **second largest** number in interval $[l, r]$.

You want to determine the position of number n in no more than $\lceil 1.5 \log_2 n \rceil$ queries. Also, since we are not clever enough, our interactor can only find the second largest number in $\Theta(r - l)$, so the sum of $r - l + 1$ over your queries should not exceed $3n$.

In this problem, the interactor is non-adaptive. That is, the permutation is fixed before all queries.

Input

The first line contains an integer T ($1 \le T \le 10000$), representing the number of test cases.

For each test case, the first line contains an integer n ($2 \le n \le 10^6$), representing the length of the permutation. It is guaranteed that the sum of n over all test cases does not exceed 10^6.

Interaction Protocol

To ask a question, print a line of the form "`? l r`" ($1 \le l < r \le n$). Then you should read the response from standard input.

To report the answer, print a line of the form "`! x`", representing that the position of number n is x.

After printing the answer, your program should process the next test case, or terminate if there are no more test cases.

After printing each line, do not forget to output end of line and flush the output. To do the latter, you can use `fflush(stdout)` or `cout.flush()` in C++, `System.out.flush()` in Java, or `stdout.flush()` in Python.

Example

standard input	standard output
2 5 3 3 3 2 2	 ? 1 5 ? 1 3 ? 2 3 ! 2 ? 1 2 ! 1

Note

Please note that the examples are only for showing the correct format of the interaction process, but you are not guaranteed to obtain a definitive result after these interactions.