样品检测

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

这是一个交互式问题。你必须在每行打印后使用 `flush` 操作。例如，在 C++ 中，你应该使用函数 `fflush(stdout)`，在 Java 中使用 `System.out.flush()`，在 Pascal 中使用 `flush(output)`，在 Python 中使用 `sys.stdout.flush()`。

某工厂生产了一批样品，编号从 1 到 n。

质检部门已经确认，其中恰好有一个样品是合格样品，恰好有一个样品是污染样品，并且这两个样品的编号不同。你并不知道它们分别是哪一个。

实验室中有一台检测仪。每次你可以选择若干个样品放入检测仪中。设你选择的样品集合为 S，检测仪会返回一个结果：`YES` 表示当且仅当合格样品在 S 中，且污染样品不在 S 中，否则返回 `NO`。

你需要在不超过 **29** 次询问内，确定合格样品和污染样品的编号。

请注意，交互器是自适应的，这意味着答案不一定是预先确定好的，交互器可以任意选择，但答案不会在交互过程中与之前返回的结果产生矛盾。

Input

输入的第一行包含一个正整数 n $(2 \le n \le 1000)$ — 表示样品的数量。

Interaction Protocol

你可以进行至多 **29** 次询问。每次询问输出一行以 `?` 开头，之后跟一个空格，然后输出一个整数 k $(0 \le k \le n)$ 表示本次放入检测仪的样品数量，接下来输出 k 个整数 $a_1, a_2, ..., a_k$ $(1 \le a_i \le n)$ — a_i 表示第 i 个被放入检测仪的样品编号。所有 a_i 必须两两不同。各项之间用单个空格分隔。

若合格样品在你询问的集合中，且污染样品不在该集合中，则交互器返回 `YES`；否则返回 `NO`。

当你确定答案后，输出一行以 `!` 开头，之后跟一个空格，然后输出两个整数 g, b $(1 \le g, b \le n,\ g \ne b)$，第一个整数表示合格样品编号，第二个整数表示污染样品编号。各项之间用单个空格分隔。

输出答案后，你的程序必须立即结束。

Example

standard input	standard output
5 YES NO YES	? 4 1 2 3 4 ? 2 3 5 ? 1 3 ! 3 5

Note

样例中 3 是合格样品编号，5 是污染样品编号。

第一次询问的集合为 $\{1, 2, 3, 4\}$。它包含合格样品 3，且不包含污染样品 5，因此交互器返回 `YES`，此时可以确定污染样品编号是 5。

第二次询问的集合为 $\{3, 5\}$。它同时包含合格样品 3 和污染样品 5，因此交互器返回 `NO`。

第三次询问的集合为 $\{3\}$。它包含合格样品 3，且不包含污染样品 5，因此交互器返回 `YES`，此时可以确定合格样品编号是 3。

最终输出 ！ 3 5，表示合格样品编号为 3，污染样品编号为 5。