

---

## Задача: Duran

|                         |                              |
|-------------------------|------------------------------|
| Имя входного файла:     | <code>standart input</code>  |
| Имя выходного файла:    | <code>standart output</code> |
| Ограничение по времени: | 2 секунды                    |
| Ограничение по памяти:  | 256 мегабайт                 |

Это интерактивная задача.

Самир Дюран сидел в ресторане и размышлял о своем задании. За соседним столом сидели  $n$  человек. Дюрану предстояло узнать их имена. Сначала эта задача казалась ему простой. Он хорошо умел подслушивать чужие разговоры.

Но была одна проблема: люди для общения не использовали имена. Они присвоили друг другу номера от 1 до  $n$  и использовали номера вместо имён.

Дюран уже давно знал номер каждого человека. Но это не приносило никакой пользы.

Единственным человеком, который знал всех этих людей по именам, был хозяин ресторана. Он не скрывал этот факт и даже был готов назвать имена. Но только за соответствующую плату.

Хозяин ресторана предложил Дюрану следующую сделку: Дюран выбирает некоторое подмножество людей, называет их номера, платит один кредит. После этого хозяин ресторана называет имена этих людей. Но имена будут названы в случайном порядке.

Для Дюрана это был последний шанс выполнить задание. Но на счету у него было всего лишь 50 кредитов. Значит задать можно было не более 50 вопросов.

Помогите Самуиру Дюрану выполнить его задание. Узнайте имена всех  $n$  людей задав не более 50 вопросов или сообщите, что узнать все имена невозможно.

### Формат входных данных

На ввод программе подается одно целое число  $n$  — количество человек за столом, ( $1 \leq n \leq 512$ ).

Далее следуют ответы на запросы.

Если был сделан некорректный запрос или превышено максимальное количество запросов, то ответом на запрос будет “-1” (без кавычек). После получения такого ответа ваша программа должна немедленно завершиться.

Если был сделан запрос на получение имен, то в первой строке ответа будет содержаться одно целое число  $m$  — количество названных имен. Следующие  $m$  строк будут содержать сами имена в случайном порядке. Имена состоят только из строчных латинских букв. Длина каждого имени не превосходит 20.

Гарантируется, что нет двух человек с одинаковыми именами.

### Формат выходных данных

Запрос выглядит следующим образом.

Если вы спрашиваете имена, то в первой строке запроса выведите “?” (без кавычек), во второй строке выведите количество запрашиваемых имен. В третьей строке выведите номера людей, для которых вы хотите узнать номера. Нельзя спрашивать имя одного человека дважды в одном запросе.

Если вы хотите вывести ответ, то в первой строке запроса выведите “!” (без кавычек), в следующей строке выведите “-1” (без кавычек), если ответ получить нельзя. Иначе, выведите список имен в том порядке, в каком они пронумерованы.

После вывода ответа программа должна немедленно завершиться.

Не забывайте сбрасывать буфер вывода после вывода каждого запроса. Это делается при помощи функции “`fflush(stdout)`”; в “C++”, “`System.out.flush()`” в “Java” и “`flush(output)`”; в “Pascal”.

---

## Примеры

| standart input | standart output |
|----------------|-----------------|
| 3              | ?               |
| 3              | 3               |
| caba aba rab   | 1 2 3           |
| 2              | ?               |
| rab aba        | 2               |
| 2              | 1 3             |
| aba caba       | ?               |
| 1              | 2               |
| aba            | 1 2             |
| 1              | ?               |
| aba            | 1               |
| 1              | 1               |
| caba           | ?               |
| 1              | 1               |
| rab            | 1               |
|                | ?               |
|                | 1               |
|                | 2               |
|                | ?               |
|                | 1               |
|                | 3               |
|                | !               |
|                | aba caba rab    |

## Примечание

Имя одного человека можно спрашивать несколько раз, но не в одном запросе. Можно делать полностью одинаковые запросы.