

Взломать коллайдер

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Это интерактивная задача.

Майлз продолжает бороться с «Alchemax». Его задача в этот раз — взломать настройки коллайдера, чтобы остановить его повторный запуск.

Всего у коллайдера есть n настроек a_i с номерами от 1 до n , каждая из них является целым положительным числом. Компания предусмотрела защиту настроек от посторонних, поэтому узнать значение определенной настройки напрямую нельзя. Однако любой человек может запросить некоторое $f(x)$, равное значению настройки с номером, циклически сдвинутому от x на c вперед ($0 \leq c \leq n-1$), то есть с номером $x+c$, если $x+c \leq n$, и $x+c-n+1$ иначе.

Майлз выяснил, что на самом деле настройки идут по возрастанию, то есть что $a_{i+1} > a_i$ для всех i от 1 до $n-1$. Теперь для остановки коллайдера Майлзу обязательно нужно найти известное только сотрудникам «Alchemax» число c — сдвиг в номерах, используемый в системе доступа к настройкам.

Узнайте значение c , если вам дано число n и возможность запрашивать значения $f(x)$ не больше 42 раз.

Формат входных данных

В первой строке задано одно целое число n — количество настроек ($1 \leq n \leq 10^5$).

Также гарантируется, что все a_i — целые числа от 1 до 10^9 включительно.

Протокол взаимодействия

Взаимодействие с интерактором проходит в виде запросов со стороны вашей программы и ответов со стороны интерактора.

Вы можете не более 42 раз запросить значение $f(x)$ для $1 \leq x \leq n$. Чтобы запросить очередное значение, выведите на отдельной строке «? x », где вместо x укажите нужное число. Если вы не превысили лимит в 42 запроса вида «?», в следующей строке интерактор выведет единственное целое число $f(x)$, равное a_{x+c} при $x+c \leq n$ или $a_{x+c-n+1}$ иначе.

Для того, чтобы вывести ответ на задачу, напечатайте «! c », где вместо c должно быть значение сдвига ($0 \leq c \leq n-1$). Вывод ответа не учитывается в количестве запросов. После вывода ответа ваша программа должна завершиться с нулевым кодом возврата.

Если в какой-то момент ваша программа превышает лимит в 42 запроса, интерактор выведет на следующей строке `-1` и завершится с вердиктом `WA` (Wrong Answer). Во избежание получения вердиктов `RE`, `TL` или `IL`, прочитав ответ `-1`, ваша программа должна завершиться с нулевым кодом возврата.

Не забывайте после каждого запроса сбрасывать буфер вывода, чтобы интерактор получил ваш запрос. Это можно сделать с помощью `std::cout.flush()` в C++, `System.out.flush()` в Java и `sys.stdout.flush()` в Python, а также аналогичными командами в других языках. Если ваша программа не сбрасывает буфер вывода, она получит вердикт `TL` или `IL`.

Примеры

стандартный ввод	стандартный вывод
5	? 1
4	? 2
5	? 3
1	! 3
5	? 4
4	? 3
3	? 2
2	? 1
1	! 0