

Потеря медальона

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

Это интерактивная задача.

Перед очередным противостоянием двух непокорных героев — Дэдпула и Росомахи, Росомаха потерял свой старый медальон, дающий невероятные силы владельцу. Как только Логан обнаружил пропажу, ему сразу стало ясно — виноват Дэдпул.

Как любитель бесконечных шуток и игр, Уэйд решил испытать терпение Росомахи, спрятав медальон на координатной прямой в некоторой точке с целой **неотрицательной** координатой (известной только Дэдпулу) и предложив ему такую игру:

За каждый ход Логан может назвать одно натуральное число x . После этого Дэдпул сделает одно из двух действий:

1. Если **текущая** координата, на которой находится медальон, делится на x — Дэдпул просто ответит на запрос числом 1.
2. Если же **текущая** координата, на которой находится медальон, не делится на x , Дэдпул ответит на запрос числом 0, и **сдвинет** медальон ближе к нулю ровно на единицу (более формально — уменьшит координату на 1).

Дэдпул, считая это забавным, отвечает Логану лишь на один вопрос в минуту. Он уверен, что зрителям такое развлечение понравится, а вот Логан больше беспокоится о том, что этот артефакт попадет в руки Кассандры Новы, что случится всего через 52 минуты, и тогда ее будет уже не остановить.

Помогите Росомахе угадать **начальное** местоположение медальона, чтобы спасти мир.

Формат входных данных

Каждый тест содержит несколько наборов входных данных. Первая строка входных данных содержит одно целое число t ($1 \leq t \leq 50$) — количество наборов входных данных.

Гарантируется, что сумма ответов (начальных координат медальона) по всем наборам входных данных не превосходит $5 \cdot 10^5$.

Протокол взаимодействия

Взаимодействие с интерактором проходит в виде запросов со стороны вашей программы и ответов со стороны интерактора. Вы можете не более 52 раз сделать запрос — число x ($1 \leq x \leq 10^9$), на который получите ответ 1 или 0. Чтобы запросить очередное значение, выведите на отдельной строке «? x », где вместо x укажите нужное число. Если вы не превысили лимит в 52 запроса вида «?», в следующей строке интерактор выведет единственное целое число — 0 или 1, по правилу, описанному в условии.

Для того, чтобы вывести ответ на задачу, напечатайте «! c », где вместо c должна быть начальная координата медальона ($0 \leq c \leq 5 \cdot 10^5$). Вывод ответа не учитывается в количестве запросов. После этого вам выведется **вердикт** — 0 или 1 в случае положительного и отрицательного ответа, соответственно.

Если в какой-то момент ваша программа превышает лимит в 52 запроса, ваша программа завершится с вердиктом WA (Wrong Answer).

Важно: 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 (Time Limit) или IL (Idleness Limit).

Пример

стандартный ввод	стандартный вывод
1	? 500000
0	? 500000
0	? 500000
0	? 500000
1	! 3
1	