

Государственный переполох

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	512 мегабайт

В стране Байтландии есть ровно n городов. Страной управляет президент, а в каждом городе сидит некоторое (не обязательно одинаковое, возможно нулевое) количество министров, которое мы назовем *размером министерства* этого города. В любом отдельно взятом городе министры пронумерованы подряд целыми положительными числами, начиная с 1. В подчинении у министра находятся все министры его города с меньшими номерами. Известно, что ни в каком городе не находится более h министров.

Назовем *важностью* министра количество его подчиненных, тогда *важность* i -го министра в любом городе в точности равна $i - 1$. Чтобы члены правительства не перекладывали обязанности друг на друга, были введены следующие правила:

1. Ни один министр не знает своей важности.
2. У каждого министра есть способ связи со всеми министрами такой же важности из других городов, и только с ними.
3. У президента в каждый момент времени есть способ связи с министрами максимальной важности из каждого города, и только с ними.

Сегодня президент решил выяснить сколько всего в Байтландии министров. Для этого он может делать два типа запросов:

1. «- i x » — уменьшить размер министерства i -го города на x . В этом случае, если x больше количества министров в городе, то ничего не происходит, иначе — в отставку уходят x министров **с максимальной важностью**. Разумеется, после этого президент получает контакт нового самого важного министра города i , а министры из других городов теряют связь с ушедшими в отставку.
2. «? i » — спросить у самого важного министра города i , с каким числом министров (включая его самого) у него есть связь. Если же в городе i не осталось министров, автоответчик сообщит президенту число n . Таким образом, в любом случае, президент узнает в ответ количество городов, в которых **на данный момент** есть хотя бы столько же министров, сколько и в i -м, включая и сам i -й город.

Помогите президенту определить, сколько в Байтландии министров. Даже если их количество уменьшится в процессе выяснения ответа, президент хочет знать, сколько их было изначально. Разумеется, время президента очень ценно, поэтому он успеет сделать не более q описанных выше запросов.

Формат входных данных

Это интерактивная задача. Помимо этого, каждый тест состоит из нескольких наборов данных.

В первой строке ввода дано целое число t — количество наборов данных в тесте ($1 \leq t \leq 5$). Гарантируется, что во всех тестах, кроме примера, $t = 5$.

Во второй строке ввода через пробел даны целые числа n , h и q — количество городов, ограничение на количество министров в городе и ограничение на число запросов ($1 \leq n \leq 5000$; $1 \leq h \leq 1\,000\,000$; $1 \leq q \leq 24\,000$). Эти ограничения являются общими для всех наборов данных текущего теста.

Далее t раз запускается протокол взаимодействия с интерактором.

Формат выходных данных

Когда ваша программа готова дать ответ на задачу, следует вывести «! a » (без кавычек), где a — предполагаемый ответ. После этого программа должна перейти к следующему набору данных или завершиться в соответствии с описанными в следующей секции правилами.

Протокол взаимодействия

Интерактор ожидает от вашей программы запросы трех типов: «- i x », «? i » и «! a », где i , x и a — целые числа ($1 \leq i \leq n$; $0 \leq x \leq h$). После каждого запроса должен следовать перевод строки. При несоблюдении вашей программой формата запросов, ваше решение может получить произвольный вердикт (отличный от ОК).

На запрос первого типа интерактор на новой строке отвечает «ОК», если операция прошла успешно, и «FAIL», если в i -м городе было меньше x министров. На запрос второго типа интерактор также на новой строке ответит числом, равным количеству городов, в которых не меньше министров, чем в i -м городе.

Запрос третьего типа означает, что ваша программа готова дать ответ на задачу. Если a — верный ответ, интерактор выведет «ОК» и перейдет к следующему тесту данного мультитеста или завершится, если такого нет. Если ответ неверен, интерактор выводит «-1» и завершается с вердиктом WA.

Ваша программа может вывести не более q запросов первого или второго типа. Запрос третьего типа (ответ на тест) в этом ограничении не учитывается. При превышении данного ограничения, интерактор выводит «-1» завершается с вердиктом WA.

Обратите внимание, что завершение интерактора означает, что следующие тесты данного мультитеста будут пропущены. Чтобы не получить при этом вердикт TL или IL, при прочтении из ввода значения «-1» ваша программа должна завершить свою работу с нулевым кодом возврата.

Система оценки

В этой задаче 25 тестов, помимо теста из примера. Каждый тест оценивается независимо по указанным ниже критериям и оценивается максимум в 4 балла. Ограничения на входные данные, используемые в каждом тесте, указаны в таблице в конце этой секции.

Баллы за тест начисляются только если на каждом из t наборов данных был дан верный ответ без превышения лимита запросов. В таком случае количество баллов за каждый набор данных вычисляется по формуле

$$\text{score}(c, j) = \max \left(1, 4 - \left\lfloor \max \left(0, \frac{c - j}{\gamma} \right) \right\rfloor \right)$$

где c — количество итераций, сделанное вашей программой, j — количество итераций, сделанное решением жюри, а γ — отдельно заданный для каждого теста параметр.

В качестве финальной оценки за тест берется минимум из t оценок его наборов данных.

Номер теста	n	h	q	γ	Дополнительные ограничения
2	10	10	24 000	12 000	-
3	20	1000	20 000	6000	-
4	1000	20	20 000	4000	-
5	1000	100	6 000	2000	Размеры министерств в разных городах принимают не более 50 различных значений
6	5000	500	6000	5	-
7	5000	5000	15 000	100	-
8	1800	20 000	24 000	250	Размеры министерств любых двух разных городов отличаются хотя бы на 5
9	100	1000	1000	300	-
10	500	1 000 000	10 000	2500	-
11	300	800 000	4000	600	Среднее геометрическое размеров министерств не превосходит 20
12	350	1 000 000	5000	700	Среднее геометрическое размеров министерств не превосходит 35
13	100	200	380	1	-
14	100	5000	700	10	-
15	2000	500 000	10 000	100	-
16	100	10 000	1000	50	-
17	2000	500 000	10 500	150	-
18	2000	1 000 000	10 030	50	-
19	2000	1 000 000	10 030	20	-
20	100	1 000 000	1820	5	-
21	200	1 000 000	3420	15	-
22	200	1 000 000	3420	5	-
23	50	1 000 000	1000	30	-
24	360	1 000 000	6500	20	-
25	720	1 000 000	12 000	30	-
26	900	1 000 000	14 000	40	-

Пример

стандартный ввод	стандартный вывод
1	
3 3 6	
	? 1
3	
	- 1 1
OK	
	- 2 3
OK	
	? 1
3	
	- 3 3
FAIL	
	- 3 2
OK	
	! 6
OK	

Замечание

Это интерактивная задача. При использовании буферизованного вывода не забывайте сбрасывать буфер при выводе запросов (`sys.stdout.flush()` в Python, `System.out().flush()` в Java и `std::cout.flush()` в C++).

В примере из условия происходят следующие действия:

1. Первым запросом выясняется, что первое министерство — самое маленькое, так как во всех трех городах хотя бы столько же министров.
2. Следующими двумя действиями в отставку отправляется один министр из первого города и три из второго. Из того, что $h = 3$, и ответ интерактора на запрос «- 2 3» — это «OK», можно сделать вывод, что во втором городе было **ровно 3** министра.
3. После этого выясняется, что во всех городах не меньше министров, чем в первом городе. Но мы знаем, что во втором их теперь 0, а значит и в первом стало 0, то есть было 1.
4. Последними двумя запросами после неудачной попытки отправить трех министров из третьего города, и удачной — двух, мы понимаем, что их было ровно 2.

Таким образом, ответ на тест из условия — $1 + 3 + 2 = 6$. Обратите внимание также, что $q = 6$, и было сделано ровно 6 запросов вида «-» и «?», тогда как запрос «!» в это количество не входит.