

Содержание

Задача A. Count Online [0.5 sec, 256 mb]	2
Задача B. Приказы [0.5 sec, 256 mb]	3
Задача C. Обратная инверсия-2 [0.5 sec, 256 mb]	4
Задача D. Gyakkyou Burai Kaiji [0.5 sec, 256 mb]	5

Задача A. Count Online [0.5 sec, 256 mb]

Вам дано множество точек на плоскости.

Нужно уметь отвечать на два типа запросов:

○ $? \ x_1 \ y_1 \ x_2 \ y_2$ — сказать, сколько точек лежит в прямоугольнике $[x_1..x_2] \times [y_1..y_2]$. Точки на границе и в углах тоже считаются. $x_1 \leq x_2, y_1 \leq y_2$.

○ $+ \ x \ y$ — добавить в множество точку $(x + \text{res} \% 100, y + \text{res} \% 101)$. Где res — ответ на последний запрос вида $?$, а $\%$ — операция взятия по модулю.

Формат входных данных

Число точек N ($1 \leq N \leq 50\,000$). Далее N точек. Число запросов Q ($1 \leq Q \leq 100\,000$). Далее Q запросов. Все координаты от 0 до 10^9 .

Формат выходных данных

Для каждого запроса GET одно целое число — количество точек внутри прямоугольника.

Пример

stdin	stdout
5	3
0 0	3
1 0	1
0 1	0
1 1	0
1 1	3
9	
? 0 1 1 2	
+ 1 2	
+ 2 2	
? 1 0 2 2	
? 0 0 0 0	
+ 3 3	
? 3 3 3 3	
? 4 3 4 3	
? 4 4 5 5	

Замечание

На самом деле добавлялись точки $(4, 5), (5, 5), (4, 4)$.

Задача В. Приказы [0.5 sec, 256 mb]

Вася работает в НИИГСД (НИИ Государственных Структур Данных). Он изучает приказы правительства далёкого государства.

В том государстве все города расположены вдоль одной дороги. Они пронумерованы в порядке обхода. Изначально качество жизни в каждом из них равно нулю.

Далее последовательно издаются указы вида «уровень жизни в городах с i по j должен стать не меньше x ».

Также есть некоторые официальные заявления. Они имеют следующую форму: «средний уровень жизни в городах с i по j равен x ». Вася нуждается в помощи с проверкой этих утверждений: для каждого из них известны i и j , требуется подсчитать верное значение x .

Можете считать, что каждый приказ исполняется, а также в каждый момент времени каждый город имеет минимальный неотрицательный уровень жизни, удовлетворяющий всем приказам.

Формат входных данных

Ввод состоит из одного или более тестов. Каждый тест начинается строкой с двумя целыми числами n и k — числом городов и событий, соответственно. Следующие k строк содержат по одному описанию события:

1. $\wedge i j x$ означает приказ: после этого, все города с номерами от i до j включительно должны иметь уровень жизни не менее x ($1 \leq x \leq 10^9$, $1 \leq i \leq j \leq n$).
2. $? i j$ означает официальное заявление: следует подсчитать средний уровень жизни в городах с i по j включительно ($1 \leq i \leq j \leq n$).

В конце ввода будет помещён тест с $n = k = 0$, который не требуется обрабатывать.

Сумма n по всему вводу не превысит 100 000. Сумма k по всему вводу не превысит 100 000.

Формат выходных данных

Для каждого официального заявления выведите на отдельной строке искомый средний уровень жизни в виде несократимой дроби с наименьшим возможным натуральным знаменателем. Если знаменатель равен 1, выведите вместо дроби целое число. Следуйте формату вывода, как это показано в примере.

Пример

stdin	stdout
10 10	0
? 1 10	1
\wedge 1 10 1	10
? 1 10	10
\wedge 2 3 10	5
\wedge 3 4 5	27/5
? 2 2	16/5
? 3 3	
? 4 4	
? 1 5	
? 1 10	
0 0	

Задача С. Обратная инверсия-2 [0.5 сек, 256 mb]

Таблицей инверсий для перестановки $A = (a_1, a_2, \dots, a_n)$ чисел $\{1, 2, \dots, N\}$ называется массив $X = (x_i)_{1 \leq i \leq N}$, в котором на i -м месте стоит количество элементов, больших i , но стоящих левее, чем i , т.е. $x_i = \text{число таких } j', \text{ что } j' < j, a_{j'} > a_j = i$.

Например, таблицей инверсий для перестановки $(2, 5, 1, 3, 4)$ будет $(2, 0, 1, 1, 0)$, а для перестановки $(6, 1, 3, 7, 5, 4, 2) = (1, 5, 1, 3, 2, 0, 0)$.

Обратной перестановкой A^{-1} к перестановке A называется такая перестановка чисел, что на i -м месте в A^{-1} стоит номер места, на котором стоит элемент, равный i , в перестановке A .

Например, для перестановки $(2, 5, 1, 3, 4)$ обратной будет $(3, 1, 4, 5, 2)$ (т.к. 1 стоит на третьем месте, 2 — на первом, 3 — на четвертом, 4 — на пятом, а 5 — на втором), а для перестановки $(2, 7, 3, 6, 5, 1, 4)$ обратной будет $(6, 1, 3, 7, 5, 4, 2)$.

Ваша задача — по таблице инверсий перестановки A посчитать таблицу инверсий обратной перестановки A^{-1} .

Формат входных данных

Файл состоит ровно из N чисел, разделенных пробелами и переводами строки, задающих таблицу инверсий перестановки A . Число N находится в пределах от 1 до **262 144**.

Формат выходных данных

Выведите N целых чисел, разделенных пробелами — таблицу инверсий для обратной перестановки.

Пример

stdin	stdout
2 0 1 1 0	1 3 0 0 0
5 0 1 3 2 1 0	1 5 1 3 2 0 0

Задача D. Gyakkyou Burai Kaiji [0.5 sec, 256 mb]

You shouldn't let kings like myself draw twice.

Однажды, прежде чем появится здесь. Каиджи потерял все. Единственное, что у него осталось — жалкая жизнь.



Правила этой игры практически такие же. Есть N различных типов карт, все типы пронумерованы числами от 1 до N включительно. Каиджи хранит свои карты в колодах. Карты одинакового типа он кладет в одинаковые колоды, а карты разного типа в разные. Индекс каждой колоды совпадает с индексом типа карт, которые она содержит.

В любой момент времени, у Каиджи может быть от 0 до 999 999 999 карт каждого типа. Однако, сейчас игроки не могут купить, продать или обменяться картами. Таким образом, количество карт каждого типа, которое есть у Каиджи, остается одинаковым в течение всей игры. В течение хода, Каиджи может сыграть, используя только одну колоду с индексами из отрезка $[i, j]$ где i и j параметры хода.

Каиджи уже изучил поведение и стратегии всех игроков и разработал выигрышную стратегию. Теперь все, что ему надо, это быстро находить ответы к текущему типу вопроса: на ходе с параметрами i и j , какое количество карт в k -й по величине колоде среди колод, которые он использует? Помогите ему ответить на эти вопросы.

Первая строка содержит целое число N , количество типов карт ($1 \leq N \leq 450\,000$).

Вторая строка используется, чтобы сгенерировать целые числа a_i , начальное количество карт каждого типа, которое есть у Каиджи ($0 \leq a_i < 10^9$). Она содержит три целых числа a_1 , l и m . ($0 \leq a_1, l, m < 10^9$); $2 \leq i \leq N$,

$$a_i = (a_{i-1} \cdot l + m) \bmod 10^9.$$

Третья строка содержит целое число B — число противников ($1 \leq B \leq 1000$). В следующих строках описывают множество игр с отдельным противником. Каждое множество описывается десятью целыми числами. Первым идет число G — число игр, сыгранных с этим противником. Затем следуют x_1 , l_x и m_x , потом y_1 , l_y и m_y , и наконец, k_1 , l_k и m_k ($1 \leq x_1 \leq y_1 \leq N$, $1 \leq k_1 \leq y_1 - x_1 + 1$, $0 \leq l_x, m_x, l_y, m_y, l_k, m_k < 10^9$). Они используются, чтобы сгенерировать вспомогательную последовательность x_g и y_g и текущие параметры i_g , j_g и k_g для $1 \leq g \leq G$:

$$\begin{aligned} x_g &= ((i_{g-1} - 1) \cdot l_x + m_x) \bmod N + 1, & 2 \leq g \leq G \\ y_g &= ((j_{g-1} - 1) \cdot l_y + m_y) \bmod N + 1, & 2 \leq g \leq G \\ i_g &= \min(x_g, y_g), & 1 \leq g \leq G \\ j_g &= \max(x_g, y_g), & 1 \leq g \leq G \\ k_g &= (((k_{g-1} - 1) \cdot l_k + m_k) \bmod (j_g - i_g + 1)) + 1, & 2 \leq g \leq G \end{aligned}$$

Сгенерированные параметры означают, что в g -й игре с текущим противником, Каиджи хочет знать количество карт в k_g -й по величине колоде среди всех колод с индексами из отрезка $[i_g, j_g]$. Общее количество игр, сыгранных Каиджи, не превышает 600 000.

Формат выходных данных

Для каждой игры g с каждым противником b , найдите число карт в k_g -й по величине колоде, среди его колод с индексами из отрезка $[i_g, j_g]$. Выведите одно число: сумму всех этих значений.

Пример

stdin	stdout
5 1 1 1 5 1 1 0 0 3 0 0 2 0 0 1 2 0 0 5 0 0 3 0 0 1 1 0 0 5 0 0 5 0 0 1 3 0 0 3 0 0 1 0 0 1 1 0 0 4 0 0 1 0 0	15

У Каиджи есть i карт i -го типа для всех $i = 1, 2, 3, 4, 5$. Каждый тип выбирается только один раз. Таким образом ответ 15.