

Задача А. Максимум на подотрезках с добавлением на отрезке

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 0.5 секунд
Ограничение по памяти: 256 мегабайт

Реализуйте эффективную структуру данных для хранения массива и выполнения следующих операций: увеличение всех элементов данного интервала на одно и то же число; поиск максимума на интервале.

Формат входных данных

В первой строке вводится одно натуральное число N ($1 \leq N \leq 100000$) — количество чисел в массиве.

Во второй строке вводятся N чисел от 0 до 100000 — элементы массива.

В третьей строке вводится одно натуральное число M ($1 \leq M \leq 30000$) — количество запросов.

Каждая из следующих M строк представляет собой описание запроса. Сначала вводится одна буква, кодирующая вид запроса (m — найти максимум, a — увеличить все элементы на отрезке).

Следом за m вводятся два числа — левая и правая граница отрезка.

Следом за a вводятся три числа — левый и правый концы отрезка и число add , на которое нужно увеличить все элементы данного отрезка массива ($0 \leq add \leq 100000$).

Формат выходных данных

Выведите в одну строку через пробел ответы на каждый запрос m .

Пример

стандартный ввод	стандартный вывод
5 2 4 3 1 5 5 m 1 3 a 2 4 100 m 1 3 a 5 5 10 m 1 5	4 104 104

Задача В. Объединение прямоугольников

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

Дано N прямоугольников со сторонами, параллельными осям координат и вершинами в целочисленных точках. Найдите площадь их объединения.

Формат входных данных

В первой строке дано число $0 \leq N < 10^5$ - количество прямоугольников. В следующих N строках даны описания прямоугольников. Каждое описание прямоугольника — это 4 числа через пробел: $\langle x_1, y_1, x_2, y_2 \rangle$. Левый нижний угол прямоугольника имеет координаты $\langle x_1, y_1 \rangle$, правый верхний угол имеет координаты $\langle x_2, y_2 \rangle$.

$$\begin{aligned} -10^9 &\leq x_1 \leq x_2 \leq 10^9; \\ -10^9 &\leq y_1 \leq y_2 \leq 10^9 \end{aligned}$$

Формат выходных данных

Выведите одно число — площадь объединения этих прямоугольников

Примеры

стандартный ввод	стандартный вывод
2 0 0 2 2 1 3 2 4	5
3 1 1 3 5 5 2 7 4 2 4 6 7	23

Задача C. Simple 2D

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 3 секунды
Ограничение по памяти: 512 мегабайт

Дан набор точек на плоскости. Напишите структуру данных, отвечающую на запросы «количество точек в прямоугольнике».

Формат входных данных

В первой строке идёт целое число N ($N \leq 10^5$) — количество точек.

В последующих N строках идут координаты точек в формате (x_i, y_i) — числа, не превосходящие по модулю 10^9 .

Далее идёт целое число M ($M \leq 10^5$) — количество запросов.

В последующих M строках идут запросы в формате $(x_{1j}, y_{1j}, x_{2j}, y_{2j})$ — числа, не превосходящие по модулю 10^9 , задающие координаты нижнего-левого и верхнего-правого угла j -ого прямоугольника-запроса. Стороны считаем принадлежащими многоугольнику.

Формат выходных данных

Для каждого запроса выведите единственное целое число — количество точек, попавших внутрь прямоугольника.

Пример

стандартный ввод	стандартный вывод
5	3
-1 -1	5
-1 0	0
0 1	1
0 -1	1
1 -1	
5	
0 -3 3 1	
-1 -1 1 1	
3 4 5 6	
-1 -1 -1 -1	
0 0 0 5	

Задача D. Присваивание, прибавление и сумма

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 1024 мегабайта

Есть массив из n элементов, изначально заполненный нулями. Вам нужно написать структуру данных, которая обрабатывает три вида запросов:

- присвоить всем элементам на отрезке от l до $r - 1$ значение v ,
- прибавить ко всем элементам на отрезке от l до $r - 1$ число v ,
- узнать сумму на отрезке от l до $r - 1$.

Формат входных данных

Первая строка содержит два числа n и m ($1 \leq n, m \leq 100000$) — размер массива и число операций. Далее следует описание операций. Описание каждой операции имеет следующий вид:

- $1 \ l \ r \ v$ — присвоить всем элементам на отрезке от l до $r - 1$ значение v ($0 \leq l < r \leq n$, $0 \leq v \leq 10^5$).
- $2 \ l \ r \ v$ — прибавить ко всем элементам на отрезке от l до $r - 1$ число v ($0 \leq l < r \leq n$, $0 \leq v \leq 10^5$).
- $3 \ l \ r$ — узнать сумму на отрезке от l до $r - 1$ ($0 \leq l < r \leq n$).

Формат выходных данных

Для каждой операции третьего типа выведите соответствующее значение.

Примеры

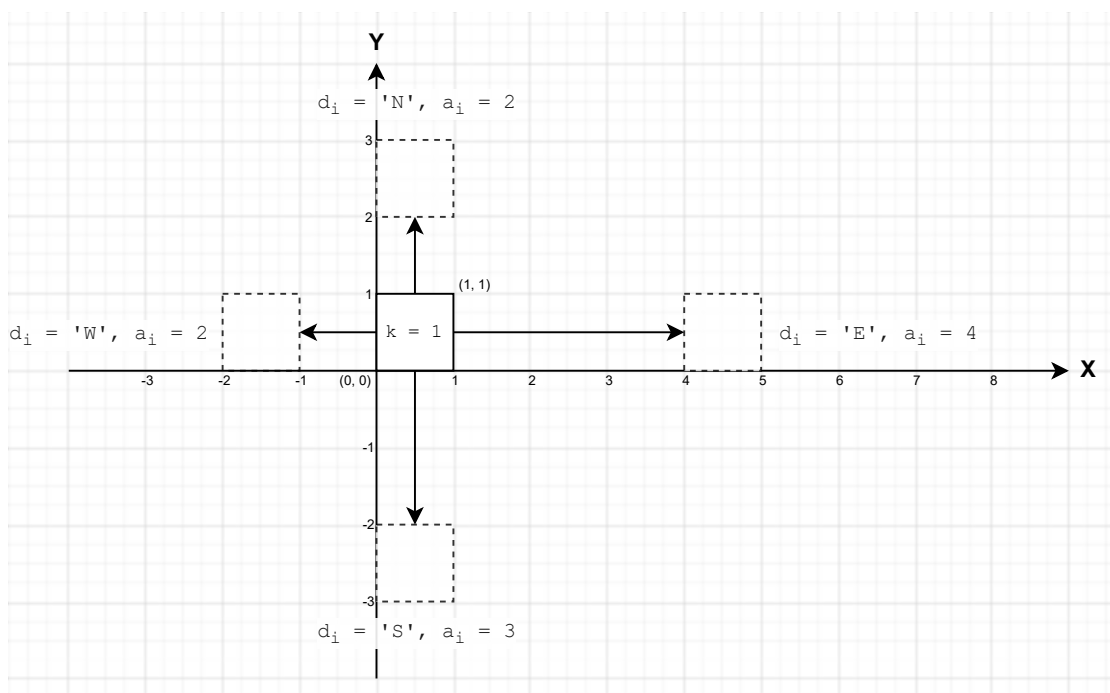
стандартный ввод	стандартный вывод
5 7	8
1 0 3 3	10
2 2 4 2	4
3 1 3	
2 1 5 1	
1 0 2 2	
3 0 3	
3 3 5	

Задача Е. Робот-пылесос

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мегабайт

Рассмотрим координатную плоскость, которую планируется очищать с использованием робота пылесоса. Робот-пылесос представляет собой квадрат размером $k \times k$ со сторонами, параллельными осям координат. Изначально левый нижний угол робота находится в точке $(0, 0)$, а правый верхний, соответственно — в точке (k, k) .

Вам дана последовательность из n перемещений робота по плоскости, i -е перемещение характеризуется направлением d_i , принимающим значения 'N' (вверх, увеличение координаты Y), 'S' (вниз, уменьшение координаты Y), 'W' (влево, уменьшение координаты X) или 'E' (вправо, увеличение координаты X), и целым числом a_i — расстоянием, на которое робот перемещается.



На рисунке приведены примеры возможных перемещений робота в каждом направлении.

Робот в каждый момент времени убирает всю площадь под собой. Иными словами, точка считается убранной тогда и только тогда, когда она в какой-то момент времени принадлежала квадрату размера $k \times k$, на котором находился робот.

По заданным перемещениям робота посчитайте суммарную площадь всей убранной поверхности.

Формат входных данных

В первой строке ввода через пробел даны два целых числа: размер робота k и количество команд n ($1 \leq k \leq 10^4$; $1 \leq n \leq 10^5$).

В i -й из следующих n строк через пробел даны направление i -го перемещения d_i и его расстояние a_i (d_i — буква 'N', 'S', 'W' или 'E'; $1 \leq a_i \leq 10^9$).

Формат выходных данных

Выведите суммарную площадь убранной роботом поверхности.

Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой подзадачи и необходимых подзадач успешно пройдены.

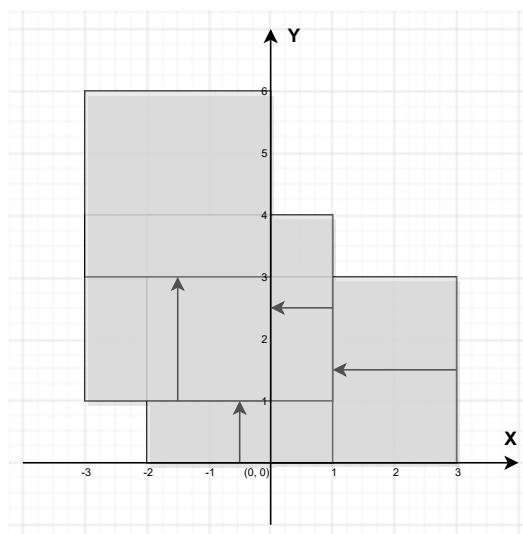
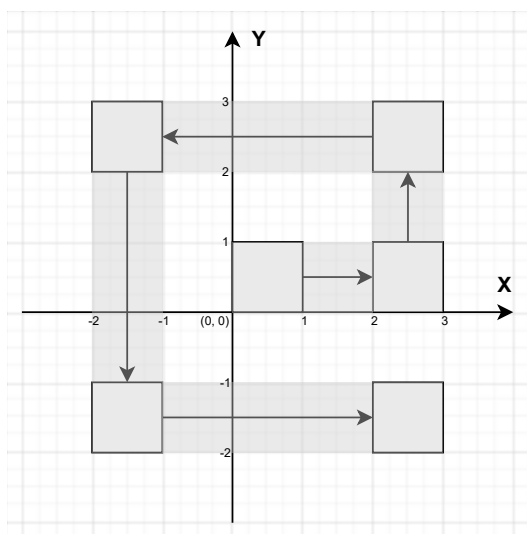
Подзадача	Баллы	Дополнительные ограничения	Необходимые подзадачи	Информация о проверке
1	9	$k = 1, n \leq 10, a_i \leq 10$		первая ошибка
2	10	$k \leq 10, n \leq 10, a_i \leq 100$	1	первая ошибка
3	11	$k \leq 1000, n \leq 1000, a_i = 1$		первая ошибка
4	8	$k \leq 10^4, n \leq 10^5, a_i = k$		первая ошибка
5	14	$k = 1, n \leq 1000, a_i \leq 10^9$	1	первая ошибка
6	15	$k \leq 10^4, n \leq 1000, a_i \leq 10^9$	1–3, 5	первая ошибка
7	16	$k = 1, n \leq 10^5, a_i \leq 10^9$	1, 5	первая ошибка
8	17	$k \leq 10^4, n \leq 10^5, a_i \leq 10^9$	1–7	первая ошибка

Примеры

стандартный ввод	стандартный вывод
1 5 E 2 N 2 W 4 S 4 E 4	17
3 4 W 2 N 1 W 1 N 2	27

Пояснение к примеру

Ниже приведены иллюстрации к перемещениям робота согласно примерам из условия. Клетки, которые робот посетил за время своих перемещений, затемнены.



Задача F. Окна

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

На экране расположены прямоугольные окна, каким-то образом перекрывающиеся (со сторонами, параллельными осям координат). Вам необходимо найти точку, которая покрыта наибольшим числом из них.

Формат входных данных

В первой строке входного файла записано число окон n ($1 \leq n \leq 50\,000$).

Следующие n строк содержат координаты окон $x_{i,1}, y_{i,1}, x_{i,2}, y_{i,2}$, где $(x_{i,1}, y_{i,1})$ — координаты левого верхнего угла i -го окна, а $(x_{i,2}, y_{i,2})$ — правого нижнего (на экране компьютера y растет сверху вниз, а x — слева направо). Все координаты — целые числа, по модулю не превосходящие 10^6 .

Формат выходных данных

В первой строке выходного файла выведите максимальное число окон, покрывающих какую-либо из точек в данной конфигурации. Во второй строке выведите два целых числа, разделенных пробелом — координаты точки, покрытой максимальным числом окон. Окна считаются замкнутыми, т. е. покрывающими свои граничные точки.

Примеры

стандартный ввод	стандартный вывод
2 0 0 3 3 1 1 4 4	2 1 3
1 0 0 1 1	1 0 1
4 0 0 1 1 0 1 1 2 1 0 2 1 1 1 2 2	4 1 1

Задача G. Ладьи и прямоугольники

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 5 секунд
Ограничение по памяти: 256 мегабайт

У Поликарпа есть шахматная доска размера $n \times m$, на которой расставлены k ладей. Поликарп еще не придумал правила игры, в которую он будет играть. Однако он уже выделил на доске q прямоугольных участков особой стратегической важности, которые должны быть надежно защищены. По мнению Поликарпа, прямоугольный участок доски надежно защищен, если все его свободные клетки бьются ладьями, стоящими на этом участке. Ладьи на остальной части доски на защиту участка не влияют. Расстановка ладей фиксирована и не может быть изменена. Напомним, что ладья бьет все клетки, расположенные с ней на одной вертикали или горизонтали, если между клеткой и ладьей нет других фигур. Помогите Поликарпу определить, все ли стратегически важные участки надежно защищены.

Формат входных данных

В первой строке содержатся четыре целых числа n , m , k и q ($1 \leq n, m \leq 100\,000$, $1 \leq k, q \leq 200\,000$) — размеры доски, количество ладей и количество стратегически важных участков. Будем считать, что клетки доски пронумерованы числами от 1 до n по горизонтали и от 1 до m по вертикали. Следующие k строк содержат пары целых чисел « $x\ y$ », описывающие положение ладей ($1 \leq x \leq n$, $1 \leq y \leq m$). Гарантируется, что все ладьи стоят в разных клетках. Следующие q строк описывают стратегически важные участки четверками чисел « $x_1\ y_1\ x_2\ y_2$ » ($1 \leq x_1 \leq x_2 \leq n$, $1 \leq y_1 \leq y_2 \leq m$). Соответствующий прямоугольный участок состоит из клеток (x, y) , для которых $x_1 \leq x \leq x_2$, $y_1 \leq y \leq y_2$. Стратегически важные участки могут пересекаться или совпадать.

Формат выходных данных

Выведите q строк. Для каждого стратегически важного участка выведите «YES», если он надежно защищен, и «NO» в противном случае.

Примеры

стандартный ввод	стандартный вывод
4 3 3 3 1 1 3 2 2 3 2 3 2 3 2 1 3 3 1 2 2 3	YES YES NO
1 1 1 1 1 1 1 1 1 1	YES

Задача Н. Перестановки

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

Вася выписал на доске в каком-то порядке все числа от 1 по N , каждое число ровно по одному разу. Количество чисел оказалось довольно большим, поэтому Вася не может окинуть взглядом все числа. Однако ему надо всё-таки представлять эту последовательность, поэтому он написал программу, которая отвечает на вопрос — сколько среди чисел, стоящих на позициях с x по y , по величине лежат в отрезке целых чисел от k до l . Сделайте то же самое.

Формат входных данных

В первой строке лежит два натуральных числа — $1 \leq N \leq 100\,000$ — количество чисел, которые выписал Вася и $1 \leq M \leq 100\,000$ — количество вопросов, которые Вася хочет задать программе. Во второй строке дано N чисел — последовательность чисел, выписанных Васей. Далее в M строках находятся описания вопросов. Каждая строка содержит четыре целых числа $1 \leq x \leq y \leq N$ и $1 \leq k \leq l \leq N$.

Формат выходных данных

Выведите M строк, каждая должна содержать единственное число — ответ на Васин вопрос.

Пример

стандартный ввод	стандартный вывод
4 2	1
1 2 3 4	3
1 2 2 3	
1 3 1 3	

Задача I. Сугробы на ЛЭП

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	0.5 секунд
Ограничение по памяти:	256 мегабайт

Служба электроснабжения проводит мониторинг уровня снега, лежащего на ЛЭП Нью-Васюки - Москва. Вся ЛЭП разбивается на участки опорами. Снег имеет свойства падать на какой-либо интервал ЛЭП, если там уже лежал какой-либо снег, то высота снежного покрова на этом участке суммируется. Также снег имеет тенденцию таять на участке трассы в результате оттепели, при этом известно, что не бывает сугробов отрицательной высоты. Энергетикам крайне важно уметь узнавать суммарную высоту снежного покрова на некоторых последовательных участках, чтобы знать вероятность обрыва проводов.

Формат входных данных

В первой строке входного файла содержатся два числа: N — ($1 \leq N \leq 10000$) и M — количество команд ($1 \leq M \leq 50000$). Каждая команда имеет вид $1\ L\ R\ S$, что означает, что на участок с L -ой опоры по R -ую опору выпало S сантиметров снега (S может быть и отрицательным, тогда это означает, что такое количества снега растаяло), или $2\ L\ R$ — запрос суммарной высоты снега на участке с L -ой опоры по R -ую. Опоры нумеруются от 0 до N . Гарантируется, что для запросов вида $1\ L\ R\ S$ при $S < 0$ на каждом участке между опорами L и R уровень снега составляет не менее S .

Формат выходных данных

На каждую команду 2 (запрос) вы должны выводить число K — суммарную высоту снежного покрова, лежащего на проводах с L -ой опоры по R -ую. Каждое число должно выводиться на новой строке. Известно, что в процессе работы суммарное количество снега на любом интервале не превышает 2^{31}

Пример

стандартный ввод	стандартный вывод
10 5	37
1 0 9 10	67
1 1 5 -3	
2 4 8	
1 0 6 25	
2 0 2	

Задача J. Простое задание

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

Это задание очень простое. Вам дана строка S длины n и q запросов, каждый запрос имеет формат $i\ j\ k$, что означает: отсортировать подстроку, состоящую из символов от i до j , в неубывающем порядке, если $k = 1$ или в невозрастающем порядке, если $k = 0$.

Выведите итоговую строку после выполнения запросов.

Формат входных данных

В первой строке записано два целых числа n, q ($1 \leq n \leq 10^5$, $0 \leq q \leq 50000$), длина строки и количество запросов, соответственно.

В следующей строке идёт сама строка S . Она состоит только из строчных английских букв.

В каждой из следующих q строк записано по три целых числа i, j, k ($1 \leq i \leq j \leq n$, $0 \leq k \leq 1$), обозначающих запрос.

Формат выходных данных

Выведите строку S после выполнения всех запросов.

Пример

стандартный ввод	стандартный вывод
10 10 ittmcsvmoa 6 7 0 2 4 0 4 10 1 1 2 0 2 9 1 5 6 1 7 9 0 1 2 0 2 6 0 3 9 1	tmacimostv

Задача К. Различные числа

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Сколько различных чисел на отрезке массива?

Формат входных данных

В первой строке дано целое число n — длина массива ($1 \leq n \leq 300\,000$).

Во второй строке дано n целых чисел a_i — элементы массива ($0 \leq a_i < 10^9$).

В третьей строке дано целое число q — количество запросов ($1 \leq q \leq 300\,000$).

i -я из следующих q строк содержит пару целых чисел l_i, r_i — границы отрезка i -го запроса ($1 \leq l_i \leq r_i \leq n$).

Формат выходных данных

Выведите ответы на запросы по одному в строке.

Пример

стандартный ввод	стандартный вывод
5	3
1 1 2 1 3	2
3	3
1 5	
2 4	
3 5	

Задача L. XOR на отрезке

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	3 секунды
Ограничение по памяти:	512 мегабайт

Вам задан массив a , состоящий из n целых чисел $a_1, a_2, \dots, a_n$. С этим массивом разрешается выполнять две операции:

1. Вычислить сумму текущих элементов массива на отрезке $[l, r]$, то есть посчитать значение $a_l + a_{l+1} + \dots + a_r$
2. Применить операцию хог с заданным числом x к каждому элементу массива на отрезке $[l, r]$, то есть выполнить $a_l = a_l \oplus x, a_{l+1} = a_{l+1} \oplus x, \dots, a_r = a_r \oplus x$. Эта операция изменяет ровно $r - l + 1$ элементов массива.

Выражение $x \oplus y$ означает применение побитовой операции хог к числам x и y .

Вам задан список из m операций указанного вида. От Вас требуется выполнить все заданные операции, для каждого запроса суммы требуется вывести полученный результат.

Формат входных данных

Входные данные В первой строке задано целое число n ($1 \leq n \leq 10^5$) - размер массива. Во второй строке через пробел заданы целые числа $a_1, a_2, \dots, a_n$ ($0 \leq a_i \leq 10^6$) - исходный массив.

В третьей строке задано целое число m ($1 \leq m \leq 10^5$) - количество операций с массивом. В i -ой из следующих m строк сперва записано целое число t_i ($1 \leq t_i \leq 2$) - тип i -го запроса. Если $t_i = 1$, то это запрос суммы, если $t_i = 2$, то это запрос на изменение элементов массива. Если i -ая операция типа 1, то далее следуют два целых числа l_i, r_i ($1 \leq l_i \leq r_i \leq n$). Если i -ая операция типа 2, то далее следуют три целых числа l_i, r_i, x_i ($1 \leq l_i \leq r_i \leq n, 1 \leq x_i \leq 10^6$). Числа в строках разделены одиночными пробелами.

Формат выходных данных

Для каждого запроса типа 1 в отдельной строке выведите сумму чисел на требуемом отрезке. Ответы на запросы выводите в том порядке, в котором они заданы во входных данных.

Примеры

стандартный ввод	стандартный вывод
5 4 10 3 13 7 8 1 2 4 2 1 3 3 1 2 4 1 3 3 2 2 5 5 1 1 5 2 1 2 10 1 2 3	26 22 0 34 11
6 4 7 4 0 7 3 5 2 2 3 8 1 1 5 2 3 5 1 2 4 5 6 1 2 3	38 28

Задача М. Фэйк Ньюз

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	3 секунды
Ограничение по памяти:	512 мегабайт

В Байтландии зарегистрировано несколько информационных агентств, в том числе новое информационное агентство ООО «неверлай дот ком». Начальник этой организации хочет выпустить сенсационное расследование. Последний месяц он ждет подходящего момента, чтобы новость стала как можно более популярной.

У каждого агентства есть свой показатель авторитетности A , который может отличаться для разных агентств. Процесс публикации новости в Байтландии устроен так: первым делом агентство ООО «неверлай дот ком» публикует новость. Это запускает цепную реакцию, устроенную следующим образом:

- Каждое агентство из тех, кто еще не опубликовал данную новость, вычисляет число P : количество других агентств, опубликовавших новость.
- Если агентство имеет показатель авторитетности A , а число уже опубликовавших новость агентств P не меньше A , то это агентство публикует данную новость.
- Процесс повторяется до тех пор, пока предыдущее условие выполняется хотя бы для одного из агентств.

ООО «неверлай дот ком» хочет выбрать момент, когда им стоит опубликовать новость, но в Байтландии часто появляются новые информационные агентства, а старые часто закрываются. От вас требуется обрабатывать запросы двух типов:

1. «+ A » появляется новое агентство с показателем авторитетности A
2. «- A » одно из агентств с показателем авторитетности A прекращает работу.

Для запросов второго типа гарантируется, что существовало хотя бы одно СМИ с показателем авторитетности A . Обратите внимание, что в один момент может существовать несколько агентств с одинаковым показателем авторитетности A .

После каждого запроса вам требуется определить, сколько агентств опубликуют новость, если ООО «неверлай дот ком» опубликует ее прямо сейчас.

Формат входных данных

В первой строке задано целое число n ($1 \leq n \leq 500\,000$) — количество запросов.

В следующих n строках вводятся запросы в формате, описанном в условии задачи: сначала задаётся один из символов «+», «-», затем следует целое число A — показатель авторитетности ($1 \leq A \leq 500\,000$).

Формат выходных данных

Выведите n строк. В каждой строке должно быть одно число — количество агентств, которые опубликуют новость от ООО «неверлай дот ком» после i -го запроса.

Примеры

стандартный ввод	стандартный вывод
5 + 4 + 1 + 3 + 2 - 3	1 2 2 5 3
8 + 1 + 1 + 2 - 1 + 4 - 1 + 1 + 1	2 3 4 3 3 1 3 5

Замечание

Рассмотрим первый пример:

- После первого запроса ни одно другое агенство не будет публиковать новость от ООО «неверлай дот ком».
- После второго запроса появилось агенство, которое готово опубликовать новость.
- После третьего запроса все еще лишь одно агенство готово опубликовать новость.
- После четвёртого запроса процесс публикации будет устроен следующим образом:
 1. ООО «неверлай дот ком» публикует новость.
 2. агенство с $A = 1$ публикует новость.
 3. агенство с $A = 2$ публикует новость.
 4. агенство с $A = 3$ публикует новость.
 5. агенство с $A = 4$ публикует новость.
- После последнего запроса процесс публикации остановится на агенстве с $A = 2$.

Задача N. JamBrains

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	3 секунды
Ограничение по памяти:	256 мегабайт

*It's not enough to write the code, you still
have to fix three hundred bugs*

*Поставь курсор в конец строки где-то в
середине кода. Нажми стрелки RUL.
Теперь повтори это много раз. Ты выучил
удобный трюк навигации.*

Костя любит есть фрикадельки с клюквенным вареньем. Особенно — когда он решает задачи глубокой ночью. На самом деле, он переел прямо перед компьютером: немного варенья пролилось из банки и испортило клавиатуру. Но задачу всё равно нужно дорешать. И, в отличие от клавиатуры, он сам застревать не хочет!

Файл с кодом, над которым работает Костя, состоит из n строк, пронумерованных от 1 до n . Строка i содержит s_i возможных позиций курсора — от $(i, 1)$ до (i, s_i) , где (i, j) означает j -ю позицию в i -й строке.

Навигация по коду со залипшими клавишами — дело непростое. К счастью, стрелки «вверх» и «вправо» всё ещё работают:

- Стрелка вверх ($\uparrow$): если курсор находится в позиции (i, j) , то если $i = 1$, он остаётся на месте; иначе перемещается в позицию $(i - 1, \min(j, s_{i-1}))$, где s_{i-1} — длина строки $i - 1$.
- Стрелка вправо ($\rightarrow$): если курсор находится в позиции (i, j) , то при $j < s_i$ он перемещается в $(i, j + 1)$; иначе — в $(i + 1, 1)$; а если он уже стоит в (n, s_n) , то остаётся на месте.

Костя использует особую технику. Он выбирает начальную позицию курсора и начинает бесконечную последовательность нажатий: u раз вверх, затем r раз вправо, затем снова u раз вверх, затем r раз вправо, и так далее.

Например, пусть $n = 5$, $u = 2$, $r = 5$, а $s = \{5, 2, 4, 3, 1\}$. Если начать из позиции $(4, 3)$, то первые нажатия переместят курсор так:

$$(4, 3) \uparrow (3, 3) \uparrow (2, 2) \rightarrow (3, 1) \rightarrow (3, 2) \rightarrow (3, 3) \rightarrow (3, 4) \rightarrow (4, 1) \dots$$

Позиция называется *успешной*, если, начав из неё и выполняя описанную бесконечную последовательность нажатий, Костя рано или поздно окажется на первой строке.

Ваша задача — помочь Косте посчитать, сколько успешных начальных позиций существует в его коде.

Однако его код ещё не закончен, поэтому нужно учитывать и q обновлений. В каждом обновлении вам дают два числа: pos и x , и вам нужно заменить значение s_{pos} на x . Нужно вывести ответ до всех обновлений и после каждого обновления.

Формат входных данных

Первая строка содержит три целых числа: n , u и r ($1 \leq n \leq 10^5$, $1 \leq u, r \leq 10^{18}$).

Вторая строка содержит n целых чисел $s_1, s_2, \dots, s_n$ ($1 \leq s_i \leq 10^{12}$): количество позиций в каждой строке.

Третья строка содержит одно целое число q ($1 \leq q \leq 10^5$) — число обновлений.

Следующие q строк содержат по два целых числа pos и x ($1 \leq pos \leq n$, $1 \leq x \leq 10^{12}$). Обновления применяются последовательно к текущему состоянию.

Формат выходных данных

Выведите $q + 1$ чисел — количество успешных начальных позиций до всех обновлений и после каждого обновления. Каждое число выводите в отдельной строке.

Примеры

стандартный ввод	стандартный вывод
5 2 5 5 2 4 3 1 1 1 1	15 11
5 2 10 5 2 4 3 1 1 2 3	11 12
10 1 42 169 42 42 42 42 42 42 42 42 42 1 2 43	211 254

Замечание

В первом примере, до всех обновлений, начав из позиции (5, 1) и выполняя «2 раза вверх, затем 5 раз вправо», получаем:

- Раунд 1: (5, 1) $\uparrow$ (4, 1) $\uparrow$ (3, 1) $\rightarrow$ (3, 2) $\rightarrow$ (3, 3) $\rightarrow$ (3, 4) $\rightarrow$ (4, 1) $\rightarrow$ (4, 2)
- Раунд 2: (4, 2) $\uparrow$ (3, 2) $\uparrow$ (2, 2) $\rightarrow$ (3, 1) $\rightarrow$ (3, 2) $\rightarrow$ (3, 3) $\rightarrow$ (3, 4) $\rightarrow$ (4, 1)
- Раунд 3: (4, 1) $\uparrow$ (3, 1) $\uparrow$ (2, 1) $\rightarrow$ (2, 2) $\rightarrow$ (3, 1) $\rightarrow$ (3, 2) $\rightarrow$ (3, 3) $\rightarrow$ (3, 4)
- Раунд 4: (3, 4) $\uparrow$ (2, 2) $\uparrow$ (1, 2) $\rightarrow \dots$

Таким образом, позиция (5, 1) является успешной.

