

Задача А. Катый ноль

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 0.3 секунд
Ограничение по памяти: 256 мегабайт

Реализуйте эффективную структуру данных, позволяющую изменять элементы массива и вычислять индекс k -го слева нуля на данном отрезке в массиве.

Формат входных данных

В первой строке вводится одно натуральное число N ($1 \leq N \leq 200\,000$) — количество чисел в массиве. Во второй строке вводятся N чисел от 0 до 100 000 — элементы массива. В третьей строке вводится одно натуральное число M ($1 \leq M \leq 200\,000$) — количество запросов. Каждая из следующих M строк представляет собой описание запроса. Сначала вводится одна буква, кодирующая вид запроса (s — вычислить индекс k -го нуля, u — обновить значение элемента). Следом за s вводится три числа — левый и правый концы отрезка и число k ($1 \leq k \leq N$). Следом за u вводятся два числа — номер элемента и его новое значение.

Формат выходных данных

Для каждого запроса s выведите результат. Все числа выводите в одну строку через пробел. Если нужного числа нулей на запрашиваемом отрезке нет, выводите -1 для данного запроса.

Пример

стандартный ввод	стандартный вывод
5	4
0 0 3 0 2	
3	
u 1 5	
u 1 0	
s 1 5 3	

Замечание

Предполагается, что из всех структур для этой задачи вы выберете дерево Фенвика. Скорее всего, любая другая структура данных получит ТЛ.

Задача В. Перестановки

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 0.75 секунд
Ограничение по памяти: 512 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 интервале от k до l . Сделайте то же самое.

Формат входных данных

В первой строке лежит два натуральных числа — $1 \leq N \leq 10^5$ — количество чисел, которые выписал Вася и $1 \leq M \leq 10^5$ — 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	

Замечание

Напиши merge-sort-tree. Другие решения будут забанены.

Задача С. Звезды

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

Вася любит наблюдать за звездами. Но следить за всем небом сразу ему тяжело. Поэтому он наблюдает только за частью пространства, ограниченной кубом размером $n \times n \times n$. Этот куб поделен на маленькие кубики размером $1 \times 1 \times 1$. Во время его наблюдений могут происходить следующие события:

1. В каком-то кубике появляются или исчезают несколько звезд.
2. К нему может заглянуть его друг Петя и поинтересоваться, сколько видно звезд в части пространства, состоящей из нескольких кубиков.

Формат входных данных

Первая строка входного файла содержит натуральное число $1 \leq n \leq 128$. Координаты кубиков — целые числа от 0 до $n - 1$. Далее следуют записи о происшедших событиях по одной в строке. В начале строки записано число m . Если m равно:

- 1, то за ним следуют 4 числа — x, y, z ($0 \leq x, y, z < N$) и k ($-20000 \leq k \leq 20000$) — координаты кубика и величина, на которую в нем изменилось количество видимых звезд;
- 2, то за ним следуют 6 чисел — $x_1, y_1, z_1, x_2, y_2, z_2$ ($0 \leq x_1 \leq x_2 < N, 0 \leq y_1 \leq y_2 < N, 0 \leq z_1 \leq z_2 < N$), которые означают, что Петя попросил подсчитать количество звезд в кубиках (x, y, z) из области: $x_1 \leq x \leq x_2, y_1 \leq y \leq y_2, z_1 \leq z \leq z_2$;
- 3, то это означает, что Васе надоело наблюдать за звездами и отвечать на вопросы Пети. Эта запись встречается во входном файле только один раз и будет последней.

Количество записей во входном файле не больше 100 002.

Формат выходных данных

Для каждого Петинго вопроса выведите искомое количество звезд.

Пример

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

Задача D. Прямоугольники

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

В связи с большим количеством покупок дачных участков, два больших, но от этого не менее гордых государства (назовем их условно «первое» и «второе»), установили ряд соглашений, касающихся участков земли около их границы.

Чтобы лучше понять нововведения, рассмотрим границу между этими государствами на карте, которая висит на стене так, что север находится сверху. Введём ортонормированную систему координат, в которой ось OX направлена с запада на восток, а OY — с юга на север. Рассмотрим n равных по величине отрезков на оси OX , i -й из этих отрезков имеет координаты $(i - 1, i)$. Каждому из них сопоставим вертикальную полосу, образованную всеми возможными прямыми, параллельными OY и проходящими через сам отрезок. Теперь, чтобы разделить государства, рассмотрим придуманную систему уровней, основанную на введённых вертикальных полосах. Для каждой полосы определим e_j уровень, который задается некоторым числом z_i . Точки, принадлежащие вертикальной полосе соответствующего отрезка, лежащие выше уровня, принадлежат первому государству, а ниже — второму.

Когда коренной житель одного из государств хочет купить прямоугольный участок земли со сторонами, параллельными осям координат (участки другого вида никого не интересуют), он может это сделать, если его родное государство доминирует на выбранном участке. Это происходит, если государство доминирует на большей, чем другое государство, части вертикальных полос, образованных отрезками на оси OX . Для вертикальных полос свойство преобладания определяется следующим образом: если площадь участка на этой полосе, принадлежащего одному из государств, строго больше площади, принадлежащей другому, то первое из них доминирует на этой полосе.

Вас просят написать программу, которая могла бы определять государство, доминирующее на участке, а также изменять границу между государствами.

Формат входных данных

В первой строке входного файла записано n — количество отрезков, на которые разделена ось OX ($1 \leq n \leq 5 \cdot 10^4$).

Во второй строке — n чисел z_i , определяющих границу между государствами ($0 \leq z_i \leq 10^9$).

В третьей строке задано m — число запросов к Вашей программе ($1 \leq m \leq 10^5$).

Далее следует m строк с запросами. Каждый запрос имеет вид « $C x z$ » или « $Q x_1 y_1 x_2 y_2$ ». Запрос вида « $C x z$ » означает, что уровень вертикальной полосы номер x стал равным z ($1 \leq x \leq n$, $1 \leq z \leq 10^9$). Запрос вида « $Q x_1 y_1 x_2 y_2$ » ($1 \leq x_1 < x_2 \leq n$, $0 \leq y_1 < y_2 \leq 10^9$) означает, что требуется вывести государство, доминирующее на участке, левой границей которого является вертикальная полоса номер x_1 (включительно), правой границей — вертикальная полоса номер x_2 (включительно), а с юга и с севера участок ограничен координатами y_1 и y_2 соответственно.

Все числа во входном файле целые.

Формат выходных данных

Для каждого запроса вида « $Q x_1 y_1 x_2 y_2$ » выведите «1», если на этом участке доминирует первое государство, «2», если второе, и «0», если ни у одного из государств преимущества нет.

Пример

стандартный ввод	стандартный вывод
2	1
0 0	1
5	0
Q 1 0 2 2	
C 1 1	
Q 1 0 2 2	
C 2 1	
Q 1 0 2 2	

Задача Е. Квиддич и футбол

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1.5 секунд
Ограничение по памяти:	256 мегабайт

Гермиона хочет посетить несколько матчей по квиддичу в Хогвартсе, а затем отправиться в Лондон и посетить несколько футбольных матчей там. Гермиона выбрала n дней в порядке времени и для каждого дня i она знает, что:

- В Хогвартсе в этот день будет матч с интересностью a_i .
- В Лондоне в этот день будет матч с интересностью b_i .

Гермиона перемещается из Хогвартса в Лондон мгновенно. (Она ведь волшебница!) Она может переместиться в Лондон в первый день, а может и не перемещаться туда совсем. Гермиона не может посетить два матча в один день.

Пусть q_k – это максимально возможная интересность матча с минимальной интересностью среди посещённых матчей в случае, если Гермиона посетит ровно k матчей. Найдите значения q_1, q_2, \dots, q_n .

Формат входных данных

Первая строка содержит целое число n ($1 \leq n \leq 2 \cdot 10^5$).

Вторая строка содержит n целых чисел a_1, a_2, \dots, a_n ($1 \leq a_i \leq 10^9$).

Третья строка содержит n целых чисел b_1, b_2, \dots, b_n ($1 \leq b_i \leq 10^9$).

Формат выходных данных

Выведите n чисел q_1, q_2, \dots, q_n .

Примеры

стандартный ввод	стандартный вывод
3 5 5 7 4 8 6	8 6 5
5 8 3 3 3 1 1 6 1 1 2	8 6 3 3 2

Задача F. Размен денег

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 4 секунды
Ограничение по памяти: 256 мегабайт

За свою долгую жизнь Боря собрал коллекцию из n монет. Он выложил все эти монеты в ряд. При этом i -я в ряду монета имеет номинал a_i .

Боря собирается в очередное путешествие, но у него осталось очень мало времени на сборы. Поэтому он хочет взять некоторый отрезок лежащих подряд монет и надеется, что ему их хватит.

Боря хочет ответить на несколько запросов. В каждом запросе Боря хочет узнать, какую минимальную сумму он не сможет заплатить без сдачи, если он возьмет все монеты с l_i -й по r_i -ю. Более формально, он хочет найти такое минимальное натуральное число z , что нельзя выбрать подмножество монет с номерами от l_i до r_i , суммарный номинал которых равен z .

Формат входных данных

В первой строке задано два целых числа n и m ($1 \leq n, m \leq 150\,000$) — количество монет у Бори и количество запросов. В следующей строке задано n чисел a_i ($1 \leq a_i \leq 10^9$) — номинал i -й монеты.

В следующих m строках задано по два числа l_i и r_i ($1 \leq l_i \leq r_i \leq n$) — описание запросов.

Формат выходных данных

На каждый из m запросов выведите минимальную сумму, которую нельзя заплатить без сдачи, воспользовавшись монетами с l_i -й по r_i -ю.

Пример

стандартный ввод	стандартный вывод
5 5	13
2 1 5 3 1	4
1 5	1
1 3	2
1 1	11
2 4	
2 5	

Задача G. Простыни

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

В один прекрасный день маленький Дональд решил вымыть N своих чистых белых простыней. После мытья он положил их на землю во дворе, чтобы их высушить. Дональд помещал простыни так, чтобы никакие две простыни не касались ни сторонами, ни углами, и чтобы их стороны не пересекались, но возможно, что он разместил меньшие простыни поверх более крупных или что простыня полностью закрывает какие-то другие простыни. Сделав это, Дональд лег спать.

Друг Дональда Ким как-то узнал о том, что Дональд сушит простыни и решил пообщаться с ним. Он нашел пейнтбольный пистолет своего отца на чердаке. Наряду с пистолетом у него было несколько пейнтбольных мячей, каждый из них имел свой цвет (не обязательно уникальный). Как только Дональд заснул, Ким вошёл во двор к Дональду и начал стрелять пр простыням из пейнтбольного пистолета. Простыни Дональда очень тонкие, поэтому, когда Ким попадает в какую-то простыню, она пропускает краску дальше, на простыню ниже (и та тоже, и так происходит, пока не закончатся простыни и краска не попадет на землю). После того, как Ким использовал все шары, он с радостью покинул двор Дональда. Дональд был очень расстроен, увидев, что случилось с его простынями. Дональд очень заинтересован в правильных данных о преступлении Кима, но он в шоке и не способен думать, поэтому просит вас сказать ему количество цветов на каждой простыне.

Мы можем представлять двор Дональда как бесконечную систему координат, а простыни - как прямоугольники, параллельные осям координат. Выстрелы Кима могут быть представлены как точки в этой системе.

Когда-то в детстве дедушка рассказал Киму, что снаряд никогда не попадает в одну воронку дважды, так что координаты всех выстрелов попарно различны.

Формат входных данных

Первая строка ввода содержит два числа - количество простынь N ($1 \leq N \leq 300000$), и количество шаров M ($1 \leq M \leq 300000$).

i -я из следующих N строк содержит четыре числа: координаты нижнего левого угла A_i, B_i ($1 \leq A_i, B_i \leq 10^9$) и верхнего правого угла C_i, D_i , ($1 \leq C_i, D_i \leq 10^9$) i -й простыни.

j -я из следующих M строк содержит три числа, где X_j, Y_j ($1 \leq X_j, Y_j \leq 10^9$) - координаты j -го выстрела Кима и K_j ($1 \leq K_j \leq 10^9$) - цвет j -го пейнтбольного шара.

Формат выходных данных

i -я из N выходных строк должна содержать количество цветов на i -м листе.

Примеры

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

Задача Н. Шахматные баталии

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

Ильдар и Ваня устали постоянно играть в шахматы, поэтому они придумали новую шахматную игру.

Игра происходит на шахматном поле размером $2n \times 2m$. Это поле имеет $2n$ строк и $2m$ столбцов. Клетки этого поля покрашены в черный и белый цвета шахматной раскраской. Более точно, клетка в i -й строке и j -м столбце имеет белый цвет, если $i + j$ чётно, и чёрный цвет в противном случае.

Игра устроена следующим образом. Ильдар помечает некоторые **белые** клетки поля как недоступные. После этого он предлагает Ване попробовать решить следующую задачу: может ли он на доступных **белых** клетках поля расставить $n \times m$ шахматных королей так, что никакие два выставленных короля не бьют друг друга, то есть не стоят на клетках поля, соседних по стороне или углу.

Конечно, Ильдар планирует сделать игру интересной и предложить Ване несколько сложных комбинаций недоступных клеток. Для этого он попросил у вас помощи. Чтобы перед игрой понять, как лучше всего действовать, он хочет потренироваться. Для этого, он берёт пустое поле и хочет q раз либо сделать какую-то доступную клетку недоступной, либо сделать какую-то недоступную клетку доступной. После каждого изменения он бы хотел знать, каким будет ответ на задачу для Вани для текущего множества недоступных клеток.

Помогите Ильдару сделать игру интересной! Напишите программу, которая будет отвечать на его запросы.

Формат входных данных

В первой строке находится три целых числа n, m, q ($1 \leq n, m, q \leq 200\,000$) — количество пар строк шахматной доски, количество пар столбцов шахматной доски и количество запросов.

Следующие q строк описывают запросы Ильдара. Каждая из этих строк содержит два целых числа i, j ($1 \leq i \leq 2n, 1 \leq j \leq 2m, i + j$ чётно). Если клетка (i, j) была недоступна, она становится доступной, в противном случае она становится недоступной.

Формат выходных данных

Выведите q строк. В i -й из этих строк выведите ответ на задачу для доски, полученной после i первых запросов Ильдара.

Выведите «YES», если Ваня может так расставить шахматных королей на незанятые белые клетки поля, что никакие два короля не будут бить друг друга, и «NO», иначе.

Примеры

стандартный ввод	стандартный вывод
1 3 3	YES
1 1	YES
1 5	NO
2 4	
3 2 10	YES
4 2	YES
6 4	NO
1 3	NO
4 2	YES
6 4	YES
2 2	NO
2 4	YES
1 3	YES
4 4	NO
3 1	

Замечание

В первом примере, после второго запроса пешки будут стоять на клетках (1, 1) и (1, 5). Тогда Ваня может поставить три короля на клетки (2, 2), (2, 4) и (2, 6).

После третьего запроса пешки будут стоять на клетках (1, 1), (1, 5) и (2, 4). Тогда отстаетя всего три пустые клетки (2, 2), (1, 3) и (2, 6). Ваня не может поставить трех королей на эти клетки, потому что короли в клетках (2, 2) и (1, 3) бьют друг друга, так как эти клетки соседние по углу.

Задача I. Медианный горный хребет

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

Берляндия — огромная страна с разнообразной географией. Одной из самых знаменитых природных достопримечательностей Берляндии является «Медианный горный хребет». Этот горный хребет представляет из себя n подряд идущих горных вершин, расположенных на одной прямой, пронумерованных в порядке следования от 1 до n . Высота i -й горной вершины равна a_i .

«Медианный горный хребет» знаменит тем, что с ним ежедневно происходит так называемое *выравнивание горных вершин*. В момент выравнивания одновременно для каждой вершины от 2 до $n - 1$ её высота становится равна медианной высоте среди неё и двух соседних гор. А именно, если до выравнивания высоты были равны b_i , то новые высоты a_i устроены следующим образом: $a_1 = b_1$, $a_n = b_n$, а для всех i от 2 до $n - 1$ $a_i = \text{median}(b_{i-1}, b_i, b_{i+1})$. Медианой трёх чисел называется второе по счёту число, если отсортировать эти три числа по возрастанию. Например, $\text{median}(5, 1, 2) = 2$, а $\text{median}(4, 2, 4) = 4$.

Недавно Берляндские учёные доказали, что какими бы ни были высоты гор, процесс выравнивания рано или поздно стабилизируется, то есть в какой-то момент высоты гор перестанут изменяться после выравнивания. Правительство Берляндии хочет понять через сколько лет это произойдёт, то есть, найти величину s — сколько произойдет выравниваний, при которых у хотя бы одной горы изменится её высота. Помогите ученым решить эту важную задачу!

Обратите внимание, что в некоторых группах тестов помимо значения s вам необходимо определить также и высоты гор после s выравниваний, то есть узнать, какими высоты гор останутся навсегда.

Формат входных данных

Первая строка содержит целые числа n и t ($1 \leq n \leq 500\,000$, $0 \leq t \leq 1$) — количество гор и параметр, который определяет, необходимо ли определить итоговые высоты гор.

Вторая строка содержит целые числа $a_1, a_2, a_3, \dots, a_n$ ($1 \leq a_i \leq 10^9$) — текущие высоты гор.

Формат выходных данных

В первой строке выведите s — число выравниваний вершин, при которых высота хотя бы одной горы изменится.

Если $t = 1$, то во второй строке выведите n чисел — итоговые высоты гор после s выравниваний.

Примеры

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

Замечание

В первом примере высоты на позициях 1 и 5 не меняются. Так как медиана чисел 1, 2, 1 это 1, то на позициях 2 и 4 после первого выравнивания оказываются числа 1, и так как медиана чисел 2, 1, 2 это 2, то на позиции 3 после первого выравнивания оказывается число 2. Итого после первого выравнивания горных вершин горы имеют высоты 1, 1, 2, 1, 1. После второго выравнивания высоты становятся 1, 1, 1, 1, 1 и дальше они меняться не будут, соответственно всего было 2 меняющих высоты выравнивания.

В третьем примере после выравнивания ни у одной горы её высота не изменится и число выравниваний, при которых высоты изменятся, равно 0. Так как $t = 0$, то выводить итоговые высоты гор не нужно.

Система оценки

Тесты к этой задаче состоят из шести групп. Баллы за каждую группу ставятся только при прохождении всех тестов группы и всех тестов **необходимых** групп. **Offline-проверка** означает, что результаты тестирования вашего решения на данной группе станут доступны только после окончания соревнования.

Группа	Баллы	Дополнительные ограничения			Необх. группы	Комментарий
		n	a_i	t		
0	0	–	–	–	–	Тесты из условия.
1	19	$n \leq 1000$	–	–	0	Гарантируется, что $c \leq 10\,000$.
2	24	–	$a_i \leq 2$	–	–	
3	14	–	$a_i \leq 100$	$t = 0$	–	
4	14	–	$a_i \leq 100$	–	0, 2, 3	
5	14	–	–	$t = 0$	3	
6	15	–	–	–	0, 1, 2, 3, 4, 5	Offline-проверка.

Задача J. Нуль

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

Вам дана последовательность $a = a_1, a_2, \dots, a_N$, состоящая из 0 и 1 и последовательность $b = b_1, b_2, \dots, b_N$, состоящая из 0 и 1. Длина каждой из последовательностей равняется N .

Вы можете выполнять Q видов операций, где i -я операция имеет следующий вид:

- Замените значения $a_{l_i}, a_{l_i+1}, \dots, a_{r_i}$ на 1.

Выполните какие-то из данных операций так, чтобы минимизировать количество индексов i таких, что $a_i \neq b_i$.

Формат входных данных

В первой строке входных данных содержится целое число N ($1 \leq N \leq 200,000$) — количество элементов в последовательностях. Во второй строке содержатся N целых чисел b_i — элементы последовательности b , равные 0 или 1. В третьей строке содержится целое число Q ($1 \leq Q \leq 200,000$) — количество возможных операций. В следующих Q строках содержатся пары целых чисел l_i, r_i ($1 \leq l_i \leq r_i \leq N$) — возможные операции.

Формат выходных данных

В единственной строке выведите минимальное количество индексов i таких, что $a_i \neq b_i$ после выполнения какого-то множества операций.

Примеры

стандартный ввод	стандартный вывод
3 1 0 1 1 1 3	1
3 1 0 1 2 1 1 3 3	0
3 1 0 1 2 1 1 2 3	1
5 0 1 0 1 0 1 1 5	2
9 0 1 0 1 1 1 0 1 0 3 1 4 5 8 6 7	3
15 1 1 0 0 0 0 0 0 1 0 1 1 1 0 0 9 4 10 13 14 1 7 4 14 9 11 2 6 7 8 3 12 7 13	5
10 0 0 0 1 0 0 1 1 1 0 7 1 4 2 5 1 3 6 7 9 9 1 5 7 9	1

Задача К. Сложная задача

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	768 мегабайт

У Егора есть большой лист клетчатой бумаги и не менее большое домашнее задание по математике. Егор, как недобросовестный ученик, планирует списать домашнюю работу с сервиса Титан Бета.

Решение задачи Егора в системе Титан Бета состоит из n формул-шагов, каждую из которых он должен перенести на лист. Разумеется, формулы в листе должны идти по порядку — слева направо. Если в какой-то момент написать следующую формулу нельзя — её обязательно надо перенести на следующую строку. Но также разрешено делать переносы и в произвольном месте решения, на усмотрение ученика, однако формула должна быть написана целиком на одной строке.

Поскольку в формулах могут возникать дробные выражения, высоты формул бывают разными. Егор относится к формулам, как к бессвязному набору символов, который влезает в какой-то клетчатый прямоугольник; а именно, i -я формула занимает прямоугольник w_i клеток в ширину и h_i клеток в высоту.

Чтобы строки читались аккуратно, Егор выравнивает все формулы, записанные в одной строке, по нижней границе строки. Таким образом, высота строки равна высоте самой высокой формуле строки. Ширина же строки равна суммарной ширине формул в строке. Суммарной высотой решения назовем суммарную высоту всех строк.

Ширина листа бумаги составляет ровно m клеток. Егора заинтересовало, какую наименьшую высоту может иметь его решение? Поскольку ему надо делать домашнюю работу, то ответ на этот вопрос предстоит найти вам.

Формат входных данных

В данной задаче ввод устроен необычным образом — для тестов последней группы часть входного файла вам придется генерировать самостоятельно.

В первой строке вводятся три числа n, m, k ($1 \leq n \leq 10^7$, $1 \leq m \leq 10^{11}$, $1 \leq k \leq \min(500\,000, n)$) — общее количество формул, ширина листа, количество формул, высота и ширина для которых будут введены далее.

В следующих k строках вводится по три целых числа i_j, w_{i_j}, h_{i_j} ($1 \leq i_j \leq n$, $1 \leq w_{i_j}, h_{i_j} \leq m$). Первое число — номер одной из формул Егора. Следующие два числа — ширина и высота этой формулы соответственно.

Гарантируется, что индексы i_j возрастают с каждой следующей строкой.

В последней строке вводятся 4 числа a, b, c, d ($1 \leq a, c \leq \min(10\,000, m)$, $0 \leq b, d \leq m$) — вспомогательные параметры, которые отвечают за генерацию остальных входных данных.

Для тех формул, номера которых не были перечислены во входных данных, высоту и ширину придётся генерировать самостоятельно. А именно, для любого i , не перечисленного во входных данных, $w_i = ((w_{i-1} \cdot a + b) \bmod m) + 1$, $h_i = ((h_{i-1} \cdot c + d) \bmod m) + 1$. Гарантируется, что $i = 1$ присутствует во входных данных (соответственно, размеры всех формул определяются однозначно).

Формат выходных данных

Выведите одно число — минимальную высоту листа шириной m , в который получится уместить все формулы.

Примеры

стандартный ввод	стандартный вывод
3 3 3 1 1 2 2 2 3 3 1 3 2 2 2 2	5
4 3 1 1 2 3 1 1 1 1	7

Замечание

В первом примере из условия оптимальным вариантом размещения будет тот, при котором первая формула находится на отдельной строке, а вторая и третья — на следующей. Таким образом, суммарная высота получается равной $2 + 3 = 5$ клеткам.

Во втором примере из условия прямоугольники имеют вид $[(2, 3), (1, 2), (3, 1), (2, 3)]$. Третий прямоугольник обязательно займет целую строку. Четвертая формула, таким образом, займет свою отдельную строчку, а первая и вторая формулы сгруппируются. Получившиеся строки будут вместе занимать $3 + 1 + 3 = 7$ клеток в высоту.

Система оценки

Тесты к этой задаче состоят из восьми групп. Баллы за каждую группу ставятся только при прохождении всех тестов группы и всех тестов **необходимых** групп. **Offline-проверка** означает, что результаты тестирования вашего решения на данной группе станут доступны только после окончания соревнования.

Группа	Баллы	Дополнительные ограничения	Необх. группы	Комментарий
		n		
0	0	–	–	Тесты из условия.
1	12	$n \leq 10, n = k$	–	
2	13	$n \leq 100, n = k$	1	
3	14	$n \leq 3000, n = k$	1, 2	
4	7	$n \leq 500\,000, n = k$	–	$h_i < h_{i+1}$
5	8	$n \leq 500\,000, n = k$	–	$h_i > h_{i+1}$
6	22	$n \leq 500\,000, n = k$	1–5	
7	22	$n \leq 5 \cdot 10^6$	0–6	
8	2	–	0–7	Offline-проверка.

Задача L. Фокус с подмножествами

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

Ваня придумал интересный фокус с множеством целых чисел.

Пусть у фокусника есть множество положительных целых чисел S . Он называет некоторое положительное целое число x . Зритель должен выбрать, не показывая фокуснику, некоторое подмножество S (возможно пустое). После этого зритель называет фокуснику размер выбранного подмножества. Фокус заключается в том, что после этого фокусник отгадывает: верно ли, что сумма элементов выбранного подмножества не превосходит x . Для пустого подмножества сумма предполагается равной 0.

Ване очень понравился этот фокус, поэтому он начал готовиться к тому, чтобы показать его публике. Для этого он приготовил некоторое множество **различных** положительных целых чисел S . Конечно, Ваня хочет, чтобы фокус обязательно получился. Он называет положительное целое число x *неудачным*, если не может быть точно уверен, что фокус пройдет удачно для любого подмножества, которое выберет зритель.

Чтобы оценить, насколько хорошее множество S он выбрал, он хочет посчитать количество неудачных положительных целых чисел для него.

Также Ваня планирует протестировать различные множества S . Поэтому он просит вас написать программу, которая найдет количество неудачных положительных целых чисел для изначального множества S и для множества S после каждого изменения. Ваня сделает q изменений своего множества, каждое изменение будет одного из двух видов:

- Добавить новое число a в множество S .
- Удалить некоторое число a из множества S .

Формат входных данных

В первой строке находятся два целых числа n, q ($1 \leq n, q \leq 200\,000$) — размер изначального множества S и количество изменений.

В следующей строке находятся n **различных** целых чисел s_1, s_2, \dots, s_n ($1 \leq s_i \leq 10^{13}$) — элементы изначального множества S .

В каждой из следующих q строк находятся два целых числа t_i, a_i ($1 \leq t_i \leq 2, 1 \leq a_i \leq 10^{13}$), описывающих очередное изменение.

- Если $t_i = 1$, то это операция добавления нового числа a_i в множество S . Гарантируется, что этого числа не было в множестве S до выполнения операции.
- Если $t_i = 2$, то это операция удаления числа a_i из множества S . Гарантируется, что это число было в множестве S до выполнения операции.

Формат выходных данных

Выведите $q + 1$ строку.

В первой строке выведите количество неудачных положительных целых чисел для изначального множества S . В следующих q строках выведите количество неудачных положительных чисел для множества S после каждого изменения.

Пример

стандартный ввод	стандартный вывод
3 11	4
1 2 3	1
2 1	6
1 5	12
1 6	19
1 7	13
2 6	8
2 2	2
2 3	10
1 10	3
2 5	0
2 7	0
2 10	

Замечание

В первом тесте изначальное $S = \{1, 2, 3\}$. Для этого множества фокус может не получиться при $x \in \{1, 2, 3, 4\}$. Например, если $x = 4$, то зритель может загадать подмножество $\{1, 2\}$, сумма элементов которого равна $3 \leq x$, а может загадать подмножество $\{2, 3\}$, сумма элементов которого равна $5 > x$. Однако в обоих случаях зритель назовет фокуснику размер подмножества 2, поэтому он не сможет точно сделать правильный ответ. При этом поскольку подмножество размера 3 единственно, а сумма в любом подмножестве меньшего размера не превосходит 5, все $x \geq 5$ не являются неудачными.

Система оценки

Тесты к этой задаче состоят из семи групп. Баллы за каждую группу ставятся только при прохождении всех тестов группы и всех тестов всех необходимых групп. **Offline-проверка** означает, что результаты тестирования вашего решения на данной группе станут доступны только после окончания соревнования.

Группа	Баллы	Ограничения			Необх. группы	Комментарий
		n	q	Доп. ограничения		
0	0	—	—	—	—	Тесты из условия.
1	10	$n \leq 12$	$q \leq 100$	все $ S \leq 12$	0	
2	10	$n \leq 100$	$q \leq 100$	$s_i, a \leq 100$	0	
3	20	$n \leq 2000$	$q \leq 2000$	—	0 – 2	
4	15	$n \leq 50\,000$	$q \leq 50\,000$	—	0 – 3	
5	15	$n \leq 100\,000$	$q \leq 100\,000$	—	0 – 4	
6	15	$n \leq 150\,000$	$q \leq 150\,000$	—	0 – 5	Offline-проверка.
7	15	—	—	—	0 – 6	Offline-проверка.

Задача М. База данных

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

Заведующая учебной частью Байтландской школы в целях контроля успеваемости издала приказ, согласно которому каждый школьник обязан в конце каждой недели предоставлять информацию о своих оценках. Требуется предоставлять лишь свой средний балл, который по мнению завуча полностью характеризует успешность школьника в течение недели.

Родителям понравилось это нововведение, ведь теперь они могут посмотреть уровень успеваемости их ребёнка относительно других школьников. Каждую субботу ученики посылают в базу данных школы свои средние оценки, которые там сохраняются, а затем, родители учеников выполняют m запросов к данной базе данных.

Пусть u — максимальное число, находящееся в базе в данный момент времени, а $cnt(x)$ — количество чисел больше либо равных x (с повторениями), находящихся в базе. Предусмотрены четыре вида запросов к базе данных:

1. Заменить хранимые в базе числа на последовательность $(cnt(1), cnt(2), \dots, cnt(u))$.
2. Добавить в базу некоторое число x .
3. Удалить из базы одно вхождение некоторого числа x , если такое имеется.
4. Вывести количество чисел, равных данному числу x .

Родители начинают запрашивать интересующую их информацию и модернизировать данные лишь после того, как все n школьников отправят свои оценки.

К сожалению, недавно у школы кончилась лицензия на эту замечательную базу данных, поэтому пока её не продлили, на все запросы придётся отвечать вам.

Формат входных данных

Первая строка входных данных содержит два числа n и m ($1 \leq n, m \leq 200\,000$) — количество школьников и количество запросов к базе данных соответственно.

Во второй строке записаны n целых чисел g_i ($1 \leq g_i \leq 200\,000$) — значения средних оценок школьников.

Далее следуют m строк, описывающих запросы в порядке их применения. Каждое описание начинается с одной из букв «t», «a», «r» или «c», означающих запрос первого, второго, третьего или четвертого типа соответственно. Для запросов второго, третьего и четвертого типа далее следует число x_i ($1 \leq x_i \leq 200\,000$) — параметр запроса.

Формат выходных данных

Сначала выведите ответы на все запросы четвертого типа. Затем выведите все числа, находящиеся в базе после выполнения всех запросов, в порядке неубывания.

Гарантируется, что в выходных данных должно будет присутствовать хотя бы одно число.

Пример

стандартный ввод	стандартный вывод
6 8	0
4 3 3 3 6 6	2
t	3 3 5 7 7
c 4	
a 5	
a 3	
r 5	
c 2	
t	
r 3	

Замечание

Рассмотрим как меняется последовательность из примера:

1. (4, 3, 3, 3, 6, 6)
2. (6, 6, 6, 3, 2, 2)
3. (6, 6, 6, 3, 2, 2, 5)
4. (6, 6, 6, 3, 2, 2, 5, 3)
5. (6, 6, 6, 3, 2, 2, 3)
6. (7, 7, 5, 3, 3, 3)
7. (7, 7, 5, 3, 3)

Ниже предоставлены критерии оценки:

№	Баллы	Ограничения	Необх. группы	Комментарий
0	0	—	—	Тесты из условия.
1	15	$n, m, x_i \leq 200$	0	—
2	15	$n, m, x_i \leq 3000$	0 – 1	—
3	35	—	—	Запросы только 1 и 4 типа.
4	35	—	0 – 3	—

Задача N. Застройка мегаполиса

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	5 секунд
Ограничение по памяти:	512 мегабайт

В 2000 году Москва оказалась застроена до такой степени, что в черте города совсем не осталось территории, пригодной для постройки новых зданий. В поисках новых источников дохода правительство города приняло план, согласно которому все железнодорожные пути в черте города реконструируются и заменяются на подземные, а высвобожденная поверхность используется для коммерческой аренды.

Планирование будущей застройки было начато с участка Октябрьской железной дороги длиной в k метров. Так как строить здания поверх образованного туннеля — долгий и сложный процесс, было принято решение закрепить новый участок за наиболее популярными передвижными точками общепита, продающими мороженное, хот-доги, кофе и тому подобное.

Участок для строительства разделён на k сегментов одинаковой длины, последовательно пронумерованных целыми числами от 1 до k . Из n поступивших в правительство заявок на получение территории i -я претендует на сегменты с l_i по r_i , причём соответствующая точка общепита будет оказывать давление величиной p_i на соответствующий отрезок поверхности. Каждую заявку правительство либо отклонит, либо полностью удовлетворит, предоставив точке все запрошенные сегменты.

Правительство города заинтересовано в том, чтобы сдать каждый сегмент новообразованной территории в аренду хотя бы одной точке общепита. При этом, чтобы уменьшить риск обвала туннеля, было принято решение минимизировать максимальное из оказываемых давлений на каком-либо из сегментов. Обратите внимание, что не запрещается сдавать в аренду один сегмент сразу нескольким точкам общепита, но в таком случае давление, оказываемое ими на данный сегмент поверхности, суммируется.

Помогите правительству одобрить такой набор заявок, чтобы каждый сегмент был сдан в аренду хотя бы одной передвижной точке, но максимальное давление, оказываемое на туннель, было как можно меньше.

Формат входных данных

В первой строке входных данных находятся два целых числа n и k ($1 \leq n \leq 100\,000$, $1 \leq k \leq 10^9$) — количество заявок на открытие точек общепита и количество сегментов поверхности.

В последующих n строках описаны заявки, каждая из которых задаётся тремя целыми числами l_i , r_i , p_i ($1 \leq l_i \leq r_i \leq 10^9$, $1 \leq p_i \leq 10^9$), соответственно границами предприятия и давлением, которое оно оказывает на поверхность туннеля.

Формат выходных данных

Выведите наименьшее возможное максимальное давление, оказываемое на поверхность туннеля точками общественного питания, при условии, что все сегменты поверхности туннеля сданы в аренду. Если не существует способа покрыть весь участок, выведите -1 .

Примеры

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

Замечание

В первом тесте из условия оптимальное решение — принять первые две заявки, тогда максимальное давление, равное 3, будет достигаться на третьем сегменте.

Во втором тесте из условия невозможно сдать в аренду третий сегмент.

В третьем тесте из условия одним из оптимальных решений будет удовлетворить все заявки, тогда максимальное давление, равное 8, будет достигаться на четвёртом сегменте. Обратите внимание, что минимизировать или максимизировать количество удовлетворённых заявок не требуется.

В четвёртом тесте из условия оптимальное решение — удовлетворить первую и четвёртую заявки, тогда на все сегменты будет оказываться одинаковое давление, равное 1.

Задача О. Прибавление

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

Дан двумерный массив целых чисел a , состоящий из n строк и m столбцов. Изначально все числа в массиве равны нулю. Необходимо обработать q запросов следующего вида:

- 1 r_1 c_1 r_2 c_2 t ($1 \leq r_1 \leq r_2 \leq n$, $1 \leq c_1 \leq c_2 \leq m - t + 1$, $1 \leq t \leq m$). Назовём операцией *прибавления в прямоугольнике*, заданном углами (x_1, y_1) и (x_2, y_2) , где $x_1 \leq x_2$, $y_1 \leq y_2$, процедуру, в результате которой ко всем элементам массива $a_{x,y}$ для которых $x_1 \leq x \leq x_2$, $y_1 \leq y \leq y_2$, прибавляется 1.

Необходимо для всех i , таких что $0 \leq i < t$ выполнить операцию прибавления в прямоугольнике, заданном углами $(r_1, c_1 + i)$, $(r_2, c_2 + i)$.

- 2 r_1 c_1 r_2 c_2 ($1 \leq r_1 \leq r_2 \leq n$, $1 \leq c_1 \leq c_2 \leq m$). Необходимо вычислить сумму элементов $a_{x,y}$ в прямоугольнике, заданном углами (r_1, c_1) , (r_2, c_2) . Поскольку данная сумма может быть очень большой, надо вычислить остаток от её деления на 2^{31} .

Формат входных данных

В первой строке заданы три целых числа n , m и q ($1 \leq n, m \leq 10^9$, $1 \leq q \leq 100\,000$) — размеры массива и количество запросов.

В следующих q строках находятся описания запросов в формате, описанном выше.

Формат выходных данных

Для каждого запроса второго типа выведите ответ в отдельной строке. Гарантируется, что хотя бы один запрос имеет тип 2.

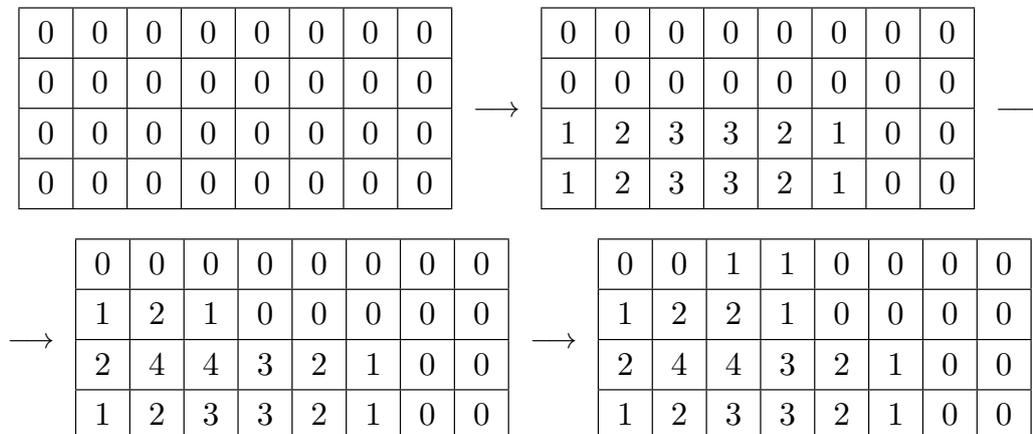
Примеры

стандартный ввод	стандартный вывод
1 4 2 1 1 1 1 2 3 2 1 1 1 4	6
4 8 6 1 1 1 2 4 3 1 2 1 3 2 2 1 3 3 4 4 1 2 3 3 4 6 2 1 2 3 5 2 1 5 4 8	5 28 6
6 6 6 1 1 1 3 5 1 1 2 1 4 4 3 2 1 1 6 5 1 2 2 6 5 1 1 2 2 5 6 1 2 2 2 3 4	48 34
13 8 8 2 2 1 8 5 1 4 2 7 8 1 2 1 2 8 4 1 3 2 13 5 2 1 4 2 8 5 3 1 2 2 9 4 1 1 4 2 9 5 2 2 3 2 7 5	0 12 130

Замечание

В первом примере мы имеем дело с одномерным массивом длины 4. Изначально массив равен $(0, 0, 0, 0)$. После первого запроса к элементам на отрезках $[1; 2]$, $[2; 3]$ и $[3; 4]$ прибавляем по 1, и получается массив $(1, 2, 2, 1)$ с суммой элементов 6.

Во втором примере в результате первых трёх запросов происходят следующие преобразования (строки нумеруются снизу вверх):



Система оценки

Тесты к этой задаче состоят из девяти групп. Баллы за каждую группу ставятся только при прохождении всех тестов группы и всех тестов **необходимых** групп.

Группа	Баллы	Доп. ограничения		Необх. группы	Комментарий
		n, m	q		
0	0	–	–	–	Тесты из условия.
1	7	$n, m \leq 100$	$q \leq 100$	0	
2	8	$n, m \leq 700$	$q \leq 700$	0 – 1	
3	10	–	$q \leq 10\,000$	–	$t_i = 1$.
4	15	–	$q \leq 10\,000$	0 – 3	
5	9	–	–	–	В запросах второго типа $(r_{i,1}, c_{i,1}) = (1, 1)$, $(r_{i,2}, c_{i,2}) = (n, m)$.
6	11	$n, m \leq 1000$	–	–	Все запросы типа 2 идут после запросов типа 1, $t_i = 1$.
7	10	$n, m \leq 1000$	–	6	Все запросы типа 2 идут после запросов типа 1.
8	14	–	–	3, 6	$t_i = 1$.
9	16	–	–	0 – 8	