

у2020-2-1. Дерево отрезков

А. Дерево отрезков на сумму

ограничение по времени на тест: 1 секунда
 ограничение по памяти на тест: 1024 мегабайта
 ввод: стандартный ввод
 вывод: стандартный вывод

В этой задаче вам нужно написать обычное дерево отрезков на сумму.

Входные данные

Первая строка содержит два числа n и m ($1 \leq n, m \leq 100000$) — размер массива и число операций. Следующая строка содержит n чисел a_i — начальное состояние массива ($0 \leq a_i \leq 10^9$). Далее следует описание операций. Описание каждой операции имеет следующий вид:

- 1 i v — присвоить элементу с индексом i значение v ($0 \leq i < n$, $0 \leq v \leq 10^9$).
- 2 l r — вычислить сумму элементов с индексами от l до $r - 1$ ($0 \leq l < r \leq n$).

Выходные данные

Для каждой операции второго типа выведите соответствующую сумму.

Пример

входные данные	Скопировать
5 5 5 4 2 3 5 2 0 3 1 1 1 2 0 3 1 3 1 2 0 5	
выходные данные	Скопировать
11 8 14	

В. Число минимумов на отрезке

ограничение по времени на тест: 1 секунда
 ограничение по памяти на тест: 1024 мегабайта
 ввод: стандартный ввод
 вывод: стандартный вывод

Теперь измените код дерева отрезков, чтобы кроме минимума на отрезке считалось также и число элементов, равных минимуму.

Входные данные

Первая строка содержит два числа n и m ($1 \leq n, m \leq 100000$) — размер массива и число операций. Следующая строка содержит n чисел a_i — начальное состояние массива ($0 \leq a_i \leq 10^9$). Далее следует описание операций. Описание каждой операции имеет следующий вид:

- 1 i v — присвоить элементу с индексом i значение v ($0 \leq i < n$, $0 \leq v \leq 10^9$).
- 2 l r — найти минимум и число элементов, равных минимуму, среди элементов с индексами от l до $r - 1$ ($0 \leq l < r \leq n$).

Выходные данные

Для каждой операции второго типа выведите два числа — минимум на заданном отрезке и число элементов, равных этому минимуму.

Пример

входные данные	Скопировать
5 5 3 4 3 5 2 2 0 3 1 1 2 2 0 3 1 0 2 2 0 5	
выходные данные	Скопировать

3 2
2 1
2 3

C. Отрезок с максимальной суммой

ограничение по времени на тест: 1 секунда
ограничение по памяти на тест: 1024 мегабайта
ввод: стандартный ввод
вывод: стандартный вывод

В этой задаче вам нужно написать дерево отрезков для нахождения подотрезка с максимальной суммой.

Входные данные

Первая строка содержит два числа n и m ($1 \leq n, m \leq 100000$) — размер массива и число операций. Следующая строка содержит n чисел a_i — начальное состояние массива ($-10^9 \leq a_i \leq 10^9$). Далее следует описание операций. Описание каждой операции имеет следующий вид: i v — присвоить элементу с индексом i значения v ($0 \leq i < n$, $-10^9 \leq v \leq 10^9$).

Выходные данные

Выведите $m + 1$ строку: максимальную сумму чисел на отрезке до всех операций и после каждой операции. Обратите внимание, что этот отрезок может быть пустым (при этом сумма на нем будет равна 0)

Примеры

входные данные	Скопировать
5 2 5 -4 4 3 -5 4 3 3 -1	
выходные данные	Скопировать
8 11 7	

входные данные	Скопировать
4 2 -2 -1 -5 -4 1 3 3 2	
выходные данные	Скопировать
0 3 3	

D. К-я единица

ограничение по времени на тест: 1 секунда
ограничение по памяти на тест: 1024 мегабайта
ввод: стандартный ввод
вывод: стандартный вывод

В этой задаче вам нужно добавить в дерево отрезков операцию нахождения k -й единицы.

Входные данные

Первая строка содержит два числа n и m ($1 \leq n, m \leq 100000$) — размер массива и число операций. Следующая строка содержит n чисел a_i — начальное состояние массива ($a_i \in \{0, 1\}$). Далее следует описание операций. Описание каждой операции имеет следующий вид:

- 1 i — изменить элемент с индексом i на противоположный.
- 2 k — найти k -ю единицу (единицы нумеруются с 0, гарантируется, что в массиве достаточное количество единиц).

Выходные данные

Для каждой операции второго типа выведите индекс соответствующей единицы (все индексы в этой задаче от 0).

Пример

входные данные	Скопировать
5 7 1 1 0 1 0 2 0 2 1 2 2 1 2 2 3	

1 0
2 0
выходные данные
0
1
3
3
1

Е. Первый элемент не меньше X - 2

ограничение по времени на тест: 1 секунда
ограничение по памяти на тест: 1024 мегабайта
ввод: стандартный ввод
вывод: стандартный вывод

В этой задаче вам нужно добавить в дерево отрезков операцию нахождения по данным x и l минимального индекса j , для которого $j \geq l$ и $a[j] \geq x$.

Входные данные
Первая строка содержит два числа n и m ($1 \leq n, m \leq 100000$) — размер массива и число операций. Следующая строка содержит n чисел a_i — начальное состояние массива ($0 \leq a_i \leq 10^9$). Далее следует описание операций. Описание каждой операции имеет следующий вид:

- 1 i v — изменить элемент с индексом i на v ($0 \leq i < n, 0 \leq v \leq 10^9$).
- 2 x l — найти минимальный индекс j , для $j \geq l$ и $a[j] \geq x$ ($0 \leq x \leq 10^9, 0 \leq l < n$). Если такого элемента нет, выведите -1 . Индексы начинаются с 0.

Выходные данные
Для каждой операции второго типа выведите ответ на запрос.

Пример
входные данные
5 7
1 3 2 4 3
2 3 0
2 3 2
1 2 5
2 4 1
2 5 4
1 3 7
2 6 1
выходные данные
1
3
2
-1
3

Ф. Прибавление и минимум

ограничение по времени на тест: 1 секунда
ограничение по памяти на тест: 1024 мегабайта
ввод: стандартный ввод
вывод: стандартный вывод

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

- прибавить к отрезку от l до $r - 1$ число v ,
- узнать минимум на отрезке от l до $r - 1$.

Входные данные
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 — прибавить значение v к отрезку от l до $r - 1$ ($0 \leq l < r \leq n, 0 \leq v \leq 10^9$).
- 2 l r — узнать минимум на отрезке от l до $r - 1$ ($0 \leq l < r \leq n$).

Выходные данные
Для каждой операции второго типа выведите соответствующее значение.

Пример

входные данные	Скопировать
5 6 1 0 3 3 2 1 2 1 1 4 4 2 1 3 2 1 4 2 3 5	
выходные данные	Скопировать
3 7 4 0	

Г. Присваивание и минимум

ограничение по времени на тест: 1 секунда

ограничение по памяти на тест: 1024 мегабайта

ввод: стандартный ввод

вывод: стандартный вывод

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

- присвоить всем элементам на отрезке от l до $r - 1$ значение v ,
- узнать минимум на отрезке от l до $r - 1$.

Входные данные

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 ($0 \leq l < r \leq n$, $0 \leq v \leq 10^9$).
- $2\ l\ r$ — узнать минимум на отрезке от l до $r - 1$ ($0 \leq l < r \leq n$).

Выходные данные

Для каждой операции второго типа выведите соответствующее значение.

Пример

входные данные	Скопировать
5 6 1 0 3 3 2 1 2 1 1 4 4 2 1 3 2 1 4 2 3 5	
выходные данные	Скопировать
3 4 4 0	

Н. Присваивание, прибавление и сумма

ограничение по времени на тест: 1 секунда

ограничение по памяти на тест: 1024 мегабайта

ввод: стандартный ввод

вывод: стандартный вывод

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 $r - 1$.

Входные данные

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 $r - 1$ ($0 \leq l < r \leq n$).

Выходные данные

Для каждой операции третьего типа выведите соответствующее значение.

Пример

входные данные	Скопировать
5 7 1 0 3 3 2 2 4 2 3 1 3 2 1 5 1 1 0 2 2 3 0 3 3 3 5	
выходные данные	Скопировать
8 10 4	

I. Криптография

ограничение по времени на тест: 2 секунды
ограничение по памяти на тест: 1024 мегабайта
ввод: стандартный ввод
вывод: стандартный вывод

Задано n матриц A_1, A_2, \dots, A_n размера 2×2 . Необходимо для нескольких запросов вычислить произведение матриц A_i, A_{i+1}, \dots, A_j . Все вычисления производятся по модулю r .

Входные данные

Первая строка входного файла содержит числа r ($1 \leq r \leq 10\,000$), n ($1 \leq n \leq 200\,000$) и m ($1 \leq m \leq 200\,000$). Следующие n блоков по две строки содержащие по два числа в строке — описания матриц. Затем следуют m пар целых чисел от 1 до n , запросы на произведение на отрезке.

Выходные данные

Выведите m блоков по две строки, по два числа в каждой — произведения на отрезках. Разделяйте блоки пустой строкой. Все вычисления производятся по модулю r .

Пример

входные данные	Скопировать
3 4 4 0 1 0 0 2 1 1 2 0 0 0 2 1 0 0 2 1 4 2 3 1 3 2 2	
выходные данные	Скопировать
0 2 0 0 0 2 0 1 0 1 0 0 2 1 1 2	

Ж. Землетрясения

ограничение по времени на тест: 1 секунда
ограничение по памяти на тест: 1024 мегабайта
ввод: стандартный ввод
вывод: стандартный вывод

Город представляет собой последовательность из n клеток, занумерованных числами от 0 до $n - 1$. Изначально все клетки пустые. Далее последовательно происходят m событий одного из двух типов:

- в клетке i строится здание с прочностью h (если в этой клетке уже было здание, оно сносится и заменяется на новое),
 - на отрезке от l до $r - 1$ случается землетрясение мощностью p , оно разрушает все здания, прочность которых не больше p .
- Ваша задача — для каждого землетрясения сказать, сколько зданий оно разрушит.

Входные данные

Первая строка содержит числа n и m — число клеток и число событий ($1 \leq n, m \leq 10^5$). Следующие m строк содержат описание событий. Описание каждого события имеет следующий вид:

- 1 i h — в клетке i строится здание с прочностью h ($0 \leq i < n$, $1 \leq h \leq 10^9$).
- 2 l r p — на отрезке от l до $r - 1$ происходит землетрясение с мощностью p ($0 \leq l < r \leq n$, $0 \leq p \leq 10^9$).

Выходные данные

Для каждого события второго типа выведите, сколько зданий было разрушено.

Пример

входные данные	Скопировать
5 9 1 0 3 1 2 5 2 0 4 3 1 1 4 1 2 7 2 1 3 6 1 3 8 1 4 4 2 0 5 10	
выходные данные	Скопировать
1 1 3	

К. Художник

ограничение по времени на тест: 2 секунды
ограничение по памяти на тест: 256 мегабайт
ввод: стандартный ввод
вывод: стандартный вывод

Итальянский художник-абстракционист Ф. Мандарино увлекся рисованием одномерных черно-белых картин. Он пытается найти оптимальное местоположение и количество черных участков картины. Для этого он проводит на прямой белые и черные отрезки, и после каждой из таких операций хочет знать количество черных отрезков на получившейся картине и их суммарную длину.

Изначально прямая — белая. Ваша задача — написать программу, которая после каждой из таких операций выводит в выходной файл интересующие художника данные.

Входные данные

В первой строке входного файла содержится общее количество нарисованных отрезков ($1 \leq n \leq 100\,000$). В последующих n строках содержится описание операций. Каждая операция описывается строкой вида $c\ x\ l$, где c — цвет отрезка (W для белых отрезков, B для черных), а сам отрезок имеет вид $[x; x + l)$, причем координаты обоих концов — целые числа, не превосходящие по модулю 500 000. Длина задается положительным целым числом.

Выходные данные

После выполнения каждой из операций необходимо вывести в выходной файл на отдельной строке количество черных отрезков на картине и их суммарную длину, разделенные одним пробелом.

Пример

входные данные	Скопировать
7 W 2 3 B 2 2 B 4 2 B 3 2	

В 7 2
W 3 1
W 0 10
выходные данные
0 0
1 2
1 4
1 4
2 6
3 5
0 0

Скопировать

L. Запросы о взвешенной сумме

ограничение по времени на тест: 1 second
ограничение по памяти на тест: 256 megabytes
ввод: стандартный ввод
вывод: стандартный вывод

В этой задаче вам надо обрабатывать запросы о взвешенной сумме для заданного массива. Формально, пусть задан массив $a[1 \dots n]$ длины n . Ваша задача уметь обрабатывать запросы двух видов:

- запрос изменения на отрезке: запрос характеризуется тремя числами l, r, d и обозначает прибавление d ко всем элементам i массива, таким что $l \leq i \leq r$,
- запрос взвешенной суммы: запрос характеризуется двумя числами l, r и обозначает вывод значения $a[l] \cdot 1 + a[l+1] \cdot 2 + \dots + a[r] \cdot (r-l+1)$.

Входные данные

В первой строке записана пара целых чисел n, m ($1 \leq n, m \leq 10^5$), n — длина массива, а m — количество запросов. Во второй строке записаны значения в массиве $a[1], a[2], \dots, a[n]$ ($-100 \leq a[i] \leq 100$). Далее в m строках записаны запросы по одному в строке. Запрос первого вида записан в форме «1 $l r d$ » ($1 \leq l \leq r \leq n, -100 \leq d \leq 100$), а запрос второго вида в форме «2 $l r$ » ($1 \leq l \leq r \leq n$).

Выходные данные

На каждый запрос второго типа выведите ответ в отдельной строке.

Примеры

входные данные
5 4
1 2 3 4 5
1 2 3 1
2 1 3
1 2 3 -1
2 1 5
выходные данные
19
55

Скопировать

Скопировать

M. Окна

ограничение по времени на тест: 2 секунды
ограничение по памяти на тест: 256 мегабайт
ввод: стандартный ввод
вывод: стандартный вывод

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

Входные данные

В первой строке входного файла записано число окон n ($1 \leq n \leq 50000$). Следующие n строк содержат координаты окон $x_{(1,i)} y_{(1,i)} x_{(2,i)} y_{(2,i)}$, где $(x_{(1,i)}, y_{(1,i)})$ — координаты левого верхнего угла i -го окна, а $(x_{(2,i)}, y_{(2,i)})$ — правого нижнего (на экране компьютера y растет сверху вниз, а x — слева направо). Все координаты — целые числа, по модулю не превосходящие $2 \cdot 10^5$.

Выходные данные

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

Примеры

входные данные
2

Скопировать

0 0 3 3 1 1 4 4	Скопировать
выходные данные	
2 1 3	

входные данные	Скопировать
1 0 0 1 1	
выходные данные	Скопировать
1 0 1	

N. Звезды

ограничение по времени на тест: 2 секунды
ограничение по памяти на тест: 256 мегабайт
ввод: стандартный ввод
вывод: стандартный вывод

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.

Входные данные

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 происходивших 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.

Выходные данные

Для каждого Петинго вопроса выведите искомое количество звезд.

Пример

входные данные	Скопировать
2 2 1 1 1 1 1 1 1 0 0 0 1 1 0 1 0 3 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	
выходные данные	Скопировать
0 1 4 2	