

Задача А. Сумма на отрезке

Имя входного файла: `sum.in`
Имя выходного файла: `sum.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Дан массив из N элементов, нужно научиться находить сумму чисел на отрезке.

Формат входных данных

Первая строка входного файла содержит два целых числа N и K — количество чисел в массиве и количество запросов ($1 \leq N \leq 100\,000$, $0 \leq K \leq 100\,000$). Следующие K строк содержат следующие запросы:

- A i x — присвоить i -му элементу массива значение x ($1 \leq i \leq n$, $0 \leq x \leq 10^9$);
- Q l r — найти сумму чисел в массиве на позициях от l до r ($1 \leq l \leq r \leq n$).

Изначально в массиве живут нули.

Формат выходных данных

На каждый запрос вида Q l r нужно вывести единственное число — сумму на отрезке.

Примеры

<code>sum.in</code>	<code>sum.out</code>
5 9	0
A 2 2	2
A 3 1	1
A 4 2	2
Q 1 1	0
Q 2 2	5
Q 3 3	
Q 4 4	
Q 5 5	
Q 1 5	

Замечание

TL для Python 4 секунды

Задача В. Мобильные телефоны

Имя входного файла: `mobile-phone.in`
Имя выходного файла: `mobile-phone.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

В области Тампере существуют станции обслуживания мобильных телефонов четвертого поколения, которые работают следующим образом. Вся область разделена на квадраты. Квадраты формируют матрицу $S \times S$, строки и столбы которой нумеруются с 0 до $S - 1$. Каждый квадрат содержит одну станцию обслуживания. Количество активных мобильных телефонов внутри квадрата может меняться, потому что телефон может перемещаться из одного квадрата в другой или выключаться. Временами каждая станция обслуживания передает сведения об изменении в количестве активных телефонов на главную станцию обслуживания (вместе с номером строки и столбца матрицы).

Напишите программу, которая принимает эти сведения и отвечает на запросы о текущем количестве активных мобильных телефонов в любой прямоугольной области.

Формат входных данных

Каждый запрос к программе задан на отдельной строке и состоит из одного целого числа, описывающего команду, и последовательности чисел, являющиеся параметрами команды.

Команда	Параметры	Значение
0	S	Инициализирует матрицу размером $S \times S$, состоящую только из нулей. Эта команда поступает один раз и является самой первой командой.
1	$X Y A$	Добавить число A к количеству активных мобильных телефонов в квадрате $(X; Y)$. Число A может быть как положительным, так и отрицательным.
2	$L B R T$	Выдать текущую сумму количеств активных мобильных телефонов во всех квадратах $(X; Y)$, таких что $L \leq X \leq R, B \leq Y \leq T$.
3	\emptyset	Означает конец входных данных. Эта команда поступает ровно один раз и является самой последней командой.

S не превосходит 1024, количество команд — 60 002, значение ячейки в любой момент времени — 2^{15} , количество активных телефонов во всей таблице — 2^{30} . Величина обновления не превосходит по модулю 2^{15} , также ни в один момент времени число активных телефонов в квадрате не становится меньше нуля.

Все индексы нумеруются начиная с нуля.

Формат выходных данных

На запросы типов 0, 1, 3 отвечать не надо. Для каждого запроса второго типа выведите строку, содержащую одно целое число — ответ на него.

Примеры

<code>mobile-phone.in</code>	<code>mobile-phone.out</code>
0 4	3
1 1 2 3	4
2 0 0 2 2	
1 1 1 2	
1 1 2 -1	
2 1 1 2 3	
3	

Задача С. Катый ноль

Имя входного файла: `kthzero.in`
Имя выходного файла: `kthzero.out`
Ограничение по времени: 1 секунда
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.

Примеры

<code>kthzero.in</code>	<code>kthzero.out</code>
5	4
0 0 3 0 2	
3	
u 1 5	
u 1 0	
s 1 5 3	

Задача D. Звезды

Имя входного файла:	stars.in
Имя выходного файла:	stars.out
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 Петиного вопроса выведите искомое количество звезд.

Примеры

stars.in	stars.out
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 Е. 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 F. Ферма

Имя входного файла: `segtree2d.in`
Имя выходного файла: `segtree2d.out`
Ограничение по времени: 10 секунды
Ограничение по памяти: 256 мегабайт

Настала весна и фермер решил заняться удобрением своего земельного участка размерами $x \times y$ метров. Для этого он закупил удобрения. До начала посевов остаётся n дней, и фермер хочет успеть сделать как можно больше.

За день фермер может одну из следующих вещей:

- увеличить продуктивность прямоугольного участка земли со сторонами, параллельными осям координат с углами (x_1, y_1) и (x_2, y_2) на значение w
- посчитать суммарную продуктивность участка $(x_1, y_1) - (x_2, y_2)$

Удобрять фермер любит сам, а вот заниматься скучными расчетами ему не интересно. Помогите ему в этом.

Формат входных данных

В первой строке входного файла записаны числа x и y ($1 \leq x, y \leq 1000$). В следующей строке написано количество оставшихся до начала посевов дней n ($1 \leq n \leq 100000$). Следующие n строк описывают действия фермера в соответственный день в следующем формате:

- 1 x_1 y_1 x_2 y_2 w — фермер удобряет участок. ($1 \leq x_1 \leq x_2 \leq x$, $1 \leq y_1 \leq y_2 \leq y$, $-10000 \leq w \leq 10000$)
- 2 x_1 y_1 x_2 y_2 — фермер просит посчитать плодородность участка. ($1 \leq x_1 \leq x_2 \leq x$, $1 \leq y_1 \leq y_2 \leq y$)

Формат выходных данных

Для каждого запроса плодородности участка в отдельной строке выведите плодородность этого участка.

Примеры

segtree2d.in	segtree2d.out
8 8	3
3	
1 2 2 8 8 2	
1 1 1 2 2 1	
2 2 2 2 2	