

Задача А. Задание котёнку

Имя входного файла: kitten.in
 Имя выходного файла: kitten.out
 Ограничение по времени: 2 секунды
 Ограничение по памяти: 64 мегабайта

Маленький котёнок заблудился в лабиринте. Дни и ночи он плутал, совсем обессилев, пока не наткнулся на Бабу-Ягу. Баба-Яга сначала хотела съесть котёнка, но потом передумала, увидев, что от него остались кожа да кости. Но котёнок все мяукал и мяукал, а топить его противно — воды Баба-Яга боится! Поэтому-то Баба-Яга и решила спровадить бедного котёнка, дав ему подробную карту лабиринта — все равно эта карта только место занимает!

Но не в правилах злодеев делать что-то, особенно доброе, просто так. Поэтому Баба-Яга дала задание котёнку — вести учет вызванных чертей. Баба-Яга хочет построить чертей в шеренгу. Шеренга будет расположена на прямой, причем черти вызываются в целых точках этой прямой. Если в какой-то точке прямой чёрт уже призван, то повторный призыв чёрта не изменяет состояния этой точки, иначе эта клетка заполняется чёртом.

Баба-Яге очень приятно восхищаться чёрным делом рук своих, и она хочет периодически задавать вопросы: а сколько, собственно, чертей находится на заданном отрезке? И этой грязной работой должен заниматься котёнок, пока Баба-Яге не надоест. Но это его единственный шанс выбраться из лабиринта!

Помогите котёнку написать программу, которая по заданной последовательности запросов Бабы-Яги и вызовов чертей ответит на каждый запрос.

Формат входного файла

Входной файл состоит из запросов Бабы-Яги и вызовов чертей. События происходят в том порядке, в котором они описаны во входном файле. Если соответствующая строка содержит одно число A_i , то оно соответствует вызову чёрта в позицию A_i . Иначе в строке содержится два числа A_i и B_i , которые означают, что Баба-Яга желала узнать, сколько же чертей заключено между точками с координатами между A_i и B_i ?

Запросов во входном файле не более 200 000, все числа A_i и B_i по модулю не превосходят 1 000 000. Последняя строка входного файла обязательно завершается переводом строки.

Формат выходного файла

Для каждого из запросов Бабы-Яги в отдельной строке необходимо вывести число — ответ на запрос.

Пример

kitten.in	kitten.out
2 3	0
5	1
2 6	1
5 6	0
7 8	2
9	2
3 9	
5	
3 9	

Задача В. Обратная инверсия-2

Имя входного файла: invers2.in
 Имя выходного файла: invers2.out
 Ограничение по времени: 2 секунды
 Ограничение по памяти: 64 мегабайта

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 входного файла

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 выходного файла

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

Пример

invers2.in	invers2.out
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 С. Звезды

Имя входного файла: stars.in
 Имя выходного файла: stars.out
 Ограничение по времени: 3 секунды
 Ограничение по памяти: 64 мегабайта

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 входного файла

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 t . Если t 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 записью.

Количество записей во входном файле не больше 100 002.

Формат выходного файла

Для каждого Петингого вопроса выведите на отдельной строке одно число — искомое количество звезд.

Пример

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	

ных 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 такой операции выводит в выходной файл интересующие художника данные.

Формат входного файла

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.

Формат выходного файла

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

painter.in	painter.out
7	0 0
W 2 3	1 2
B 2 2	1 4
B 4 2	1 4
B 3 2	2 6
B 7 2	3 5
W 3 1	0 0
W 0 10	

Задача D. Художник

Имя входного файла: painter.in
 Имя выходного файла: painter.out
 Ограничение по времени: 4 секунды
 Ограничение по памяти: 64 мегабайта

Не успев дорисовать свой гениальный футуристический шедевр, М. Калевич увлекся рисованием одномер-