

Содержание

13.base [2/2]	3
Задача 13A. Сумма [0.4 sec, 256 mb]	3
Задача 13B. Звёзды [0.1 sec, 256 mb]	4
13.advanced [3/8]	5
Задача 13C. RMQ [0.5 sec, 256 mb]	5
Задача 13D. Сумма [0.6 sec, 256 mb]	6
Задача 13E. Художник [1 sec, 256 mb]	7
Задача 13F. <i>K</i> -инверсии [0.1 sec, 256 mb]	8
Задача 13G. Перестановки [0.8 sec, 256 mb]	9
Задача 13H. Окна [0.8 sec, 256 mb]	10
Задача 13I. Вложенные отрезки [1.5 sec, 256 mb]	11
Задача 13J. Ближайшая большая справа [0.5 sec, 256 mb]	12
13.hard [0/3]	13
Задача 13K. Золотые рудники [0.2 sec, 256 mb]	13
Задача 13L. Обратная инверсия-2 [1 sec, 256 mb]	14
Задача 13M. Приказы [3 sec, 256 mb]	15

Общая информация:

Вход в констест: <http://contest.yandex.ru/contest/3539/>

Дедлайн на задачи: до 2016-12-12 23:59.

К каждой главе есть более простые задачи (base), посложнее (advanced), и сложные (hard).

В скобках к каждой главе написано сколько любых задач из этой главы нужно сдать.

Сайт курса: <https://compscicenter.ru/courses/algorithms-1/2016-autumn/>

Семинары ведут Сергей Копелиович (burunduk30@gmail.com, vk.com/burunduk1) и Алексей Кладов (aleksey.kladov@gmail.com).

В каждом условии указан таймлимит для C/C++.

Таймлимит для Java примерно в 2-3 раза больше.

Таймлимит для Python примерно в 6 раз больше.

C++:

Быстрый ввод-вывод.

http://acm.math.spbu.ru/~sk1/algo/input-output/fread_write_export.cpp.html Более подробно про ввод-вывод.

http://acm.math.spbu.ru/~sk1/algo/input-output/cpp_common.html

В некоторых задачах нужен STL, который активно использует динамическую память (set-ы, map-ы) переопределение стандартного аллокатора ускорит вашу программу:

<http://acm.math.spbu.ru/~sk1/algo/memory.cpp.html>

Java:

Быстрый ввод-вывод.

http://acm.math.spbu.ru/~sk1/algo/input-output/java/java_common.html

13.base [2/2]

Задача 13А. Сумма [0.4 sec, 256 mb]

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 — число чисел в массиве и количество запросов. ($1 \leq N \leq 100\,000$), ($0 \leq K \leq 100\,000$). Следующие K строк содержат запросы

1. A i x — присвоить i -му элементу массива значение x ($1 \leq i \leq n$, $0 \leq x \leq 10^9$)
2. 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

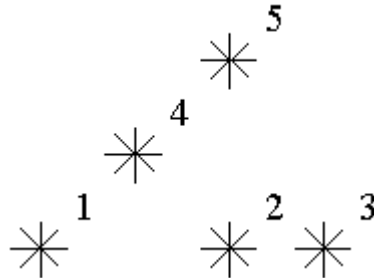
На каждый запрос вида Q l r нужно вывести единственное число — сумму на отрезке.

Примеры

sum.in	sum.out
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	

Задача 13В. Звёзды [0.1 sec, 256 mb]

Астрономы часто исследуют звёздные карты, на которых звёзды представлены точками на плоскости, каждая звезда имеет декартовы координаты. Пусть уровень звезды – количество звёзд, которые не выше и не правее данной звезды. Астрономы хотят найти распределение уровней звёзд.



Для примера посмотрим на карту звёзд на картинке выше. Уровень звезды номер 5 равен 3 (т.к. есть звёзды с номерами 1, 2, 4). Уровни звёзд 2 и 4 равны 1. На данной карте есть только одна звезда на уровне 0, две звезды на уровне 1, одна звезда на уровне 2 и одна звезда на уровне 3. Напишите программу, считающую количество звёзд на каждом уровне.

Формат входных данных

Вам дан один или несколько тестов. Каждый тест описывается следующим образом.

В первой строке количество звёзд N ($1 \leq N \leq 15\,000$). Следующие N строк описывают координаты звёзд (два целых числа X и Y , разделённые пробелом, $0 \leq X, Y \leq 32\,000$). В каждой точке плоскости находится не более одной звезды. Звёзды перечислены в порядке возрастания Y координаты, при равенстве в порядке возрастания X координаты.

Формат выходных данных

Выведите ответ для каждого теста. Ответ для теста описывается следующим образом. N строк, по одному числу в строке. i -я строка содержит количество звёзд на уровне i ($i = 0 \dots N-1$).

Примеры

stars.in	stars.out
5	1
1 1	2
5 1	1
7 1	1
3 3	0
5 5	1
5	2
1 1	1
5 1	1
7 1	0
3 3	
5 5	

13.advanced [3/8]

Задача 13С. RMQ [0.5 sec, 256 mb]

Дан массив $a[1..n]$. Требуется написать программу, обрабатывающую два типа запросов.

- Запрос “max $l r$ ”. Требуется найти максимум в массиве a от l -ой ячейки до r -ой включительно.
- Запрос “add $l r v$ ”. Требуется прибавить значение v к каждой ячейке массива a от l -ой до r -ой включительно.

Формат входных данных

Первая строка содержит два целых числа n и q ($1 \leq n, q \leq 10^5$) – длина массива и число запросов соответственно. Вторая строка содержит n целых чисел a_1, \dots, a_n ($|a_i| \leq 10^5$), задающих соответствующие значения массива. Следующие q строк содержат запросы.

В зависимости от типа запрос может иметь вид либо “max $l r$ ”, либо “add $l r v$ ”. При этом $1 \leq l \leq r \leq n$, $|v| \leq 10^5$.

Формат выходных данных

Для каждого запроса вида “max $l r$ ” требуется в отдельной строке выдать значение соответствующего максимума.

Примеры

rmq.in	rmq.out
5 3	3
1 2 3 4 -5	7
max 1 3	
add 1 2 5	
max 1 3	

Задача 13D. Сумма [0.6 сек, 256 mb]

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 — число чисел в массиве и количество запросов. ($1 \leq N \leq 100\,000$), ($0 \leq K \leq 100\,000$). Следующие K строк содержат запросы:

1. A l r x — присвоить элементам массива с позициями от l до r значение x ($1 \leq l \leq r \leq N$, $0 \leq x \leq 10^9$)
2. Q l r — найти сумму чисел в массиве на позициях от l до r . ($1 \leq l \leq r \leq N$)

Изначально массив заполнен нулями.

Формат выходных данных

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

Примеры

sum.in	sum.out
5 9	3
A 2 3 2	2
A 3 5 1	3
A 4 5 2	4
Q 1 3	2
Q 2 2	7
Q 3 4	
Q 4 5	
Q 5 5	
Q 1 5	

Задача 13Е. Художник [1 сек, 256 mb]

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 интересные художника данные.

Формат входных данных

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 выходных данных

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 13F. K -инверсии [0.1 sec, 256 mb]

Пусть дана перестановка a_1, a_2, \dots, a_n . Назовем k -инверсией набор чисел i_1, i_2, \dots, i_k таких, что $1 \leq i_1 < i_2 < \dots < i_k \leq n$ и $a_{i_1} > a_{i_2} > \dots > a_{i_k}$. Ваша задача — подсчитать количество различных k -инверсий в заданной перестановке.

Формат входных данных

В первой строке входного файла находятся число n — длина перестановки ($1 \leq n \leq 20\,000$), и число k ($2 \leq k \leq 10$). Во второй строке n чисел — сама перестановка.

Формат выходных данных

В выходной файл выведите единственное число — количество k -инверсий в заданной перестановке по модулю 10^9 .

Пример

kinverse.in	kinverse.out
3 2	2
3 1 2	
5 3	10
5 4 3 2 1	

Задача 13G. Перестановки [0.8 сек, 256 mb]

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 — $1 \leq N \leq 100\,000$ — 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

permutation.in	permutation.out
4 2	1
1 2 3 4	3
1 2 2 3	
1 3 1 3	

Задача 13Н. Окна [0.8 sec, 256 mb]

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 $x_{(1,i)} y_{(1,i)} x_{(2,i)} y_{(2,i)}$, 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 $2 \cdot 10^5$.

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, разделенные пробелом — координаты точки, покрытой максимальным числом окон. Окна считаются замкнутыми, т.е. покрывающими свои граничные точки.

Пример

windows.in	windows.out
2	2
0 0 3 3	3 2
1 1 4 4	

Задача 13I. Вложенные отрезки [1.5 сек, 256 mb]

Даны n отрезков на прямой. Пара отрезков называются вложенной, если отрезки не совпадают, и один отрезок содержит второй. Посчитать количество пар вложенных отрезков.

Формат входных данных

Целое число n ($1 \leq n \leq 300\,000$) и n пар целых чисел $0 \leq l_i \leq r_i \leq 10^9$.

Формат выходных данных

Одно число – количество пар вложенных отрезков.

Примеры

segs.in	segs.out
4	3
1 4	
2 5	
1 3	
3 4	

Система оценки

Подзадача 1 (30 баллов): $1 \leq n \leq 1000$.

Подзадача 2 (70 баллов): $1 \leq n \leq 300\,000$.

Задача 13J. Ближайшая большая справа [0.5 сек, 256 mb]

Дан массив a из n чисел. Нужно обрабатывать запросы:

0. $\text{set}(i, x) - a[i] = x$;
1. $\text{get}(i, x) -$ найти $\min k: k \geq i$ и $a_k \geq x$.

Формат входных данных

На первой строке длина массива n и количество запросов m . На второй строке n целых чисел – массив a . Следующие m строк содержат запросы.

Индексы в массиве нумеруются с 1.

Запрос типа `set`: “0 i x”.

Запрос типа `get`: “1 i x”.

$$1 \leq n, m \leq 200\,000.$$

$$1 \leq i \leq n.$$

$$0 \leq x, a_i \leq 200\,000.$$

Формат выходных данных

На каждой запрос типа `get` на отдельной строке выведите k .

Если такого k не существует, выведите -1 .

Примеры

nearandmore.in	nearandmore.out
4 5	1
1 2 3 4	3
1 1 1	-1
1 1 3	2
1 1 5	
0 2 3	
1 1 3	

13.hard [0/3]

Задача 13К. Золотые рудники [0.2 sec, 256 mb]

Байтмен, один из заслуженных работников компании по добыче золота в Байтленде, собирается в этом году на пенсию. Начальство компании решило вознаградить его за заслуги перед отечеством. Байтмену разрешили присвоить себе прямоугольную часть земли, со сторонами s и w , параллельными осям координат, со всеми входящими туда рудниками. Положение (сдвиг) участка он выбирает сам. Назовем стоимостью участка количество рудников, лежащих внутри него и на его границе. Ваша задача вычислить максимально возможную стоимость такого участка.

Формат входных данных

В первой строке входного файла находятся два целых числа s и w , разделенных одиночным пробелом ($1 \leq s, w \leq 10\,000$). Они означают длины сторон, параллельных осям Ox и Oy соответственно. Во второй строке находится целое число N ($1 \leq N \leq 15\,000$), количество рудников. В последующих N строках находятся координаты рудников. В $i+2$ строке содержатся 2 целых числа x, y ($-30\,000 \leq x, y \leq 30\,000$), означающих координаты i -го рудника.

Формат выходных данных

В первой строке выходного файла нужно вывести максимальное количество рудников внутри площадки. Во второй строке нужно вывести координату угла любой из возможных оптимальных площадок. У прямоугольника четыре угла. Вам следует выводить тот, у которого x и y координаты максимальны. Выведенные координаты должны быть целыми и не должны превосходить 10^9 по абсолютной величине.

Пример

mine.in	mine.out
1 2	4
12	3 3
0 0	
1 1	
2 2	
3 3	
4 5	
5 5	
4 2	
1 4	
0 5	
5 0	
2 3	
3 2	

Задача 13L. Обратная инверсия-2 [1 sec, 256 mb]

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 входных данных

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 выходных данных

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 13М. Приказы [3 sec, 256 mb]

Вася работает в НИИГСД (НИИ Государственных Структур Данных). Он изучает приказы правительства далёкого государства.

В том государстве все города расположены вдоль одной дороги. Они пронумерованы в порядке обхода. Изначально качество жизни в каждом из них равно нулю.

Далее последовательно издаются указы вида «уровень жизни в городах с i по j должен стать не меньше x ».

Также есть некоторые официальные заявления. Они имеют следующую форму: «средний уровень жизни в городах с i по j равен x ». Вася нуждается в помощи с проверкой этих утверждений: для каждого из них известны i и j , требуется подсчитать верное значение x .

Можете считать, что каждый приказ исполняется, а также в каждый момент времени каждый город имеет минимальный неотрицательный уровень жизни, удовлетворяющий всем приказам.

Формат входных данных

Ввод состоит из одного или более тестов. Каждый тест начинается строкой с двумя целыми числами n и k — числом городов и событий, соответственно. Следующие k строк содержат по одному описанию события:

1. $\wedge i j x$ означает приказ: после этого, все города с номерами от i до j включительно должны иметь уровень жизни не менее x ($1 \leq x \leq 10^9$, $1 \leq i \leq j \leq n$).
2. $? i j$ означает официальное заявление: следует подсчитать средний уровень жизни в городах с i по j включительно ($1 \leq i \leq j \leq n$).

В конце ввода будет помещён тест с $n = k = 0$, который не требуется обрабатывать.

Сумма n по всему вводу не превысит 100 000. Сумма k по всему вводу не превысит 100 000.

Формат выходных данных

Для каждого официального заявления выведите на отдельной строке искомый средний уровень жизни в виде несократимой дроби с наименьшим возможным натуральным знаменателем. Если знаменатель равен 1, выведите вместо дроби целое число. Следуйте формату вывода, как это показано в примере.

Пример

orders.in	orders.out
10 10	0
? 1 10	1
\wedge 1 10 1	10
? 1 10	10
\wedge 2 3 10	5
\wedge 3 4 5	27/5
? 2 2	16/5
? 3 3	
? 4 4	
? 1 5	
? 1 10	
0 0	