Двоичное дерево поиска

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

Реализуйте сбалансированное двоичное дерево поиска.

Формат входных данных

Входной файл содержит описание операций с деревом, их количество не превышает 100000. В каждой строке находится одна из следующих операций:

- `insert` x — добавить в дерево ключ x. Если ключ x в дереве уже есть, то ничего делать не надо.
- `delete` x — удалить из дерева ключ x. Если ключа x в дереве нет, то ничего делать не надо.
- `exists` x — если ключ x есть в дереве, выведите «`true`», иначе «`false`»
- `next` x — выведите минимальный элемент в дереве, строго больший x, или «`none`», если такого нет.
- `prev` x — выведите максимальный элемент в дереве, строго меньший x, или «`none`», если такого нет.

Все числа во входном файле целые и по модулю не превышают 10^9.

Формат выходных данных

Выведите последовательно результат выполнения всех операций `exists`, `next`, `prev`. Следуйте формату выходного файла из примера.

Примеры

bst.in	bst.out
insert 2 insert 5 insert 3 exists 2 exists 4 next 4 prev 4 delete 5 next 4 prev 4	true false 5 3 none 3

И снова сумма...

Имя входного файла:	`sum2.in`
Имя выходного файла:	`sum2.out`
Ограничение по времени:	3 секунды
Ограничение по памяти:	256 мегабайт

Реализуйте структуру данных, которая поддерживает множество S целых чисел, с котором разрешается производить следующие операции:

- $add(i)$ — добавить в множество S число i (если он там уже есть, то множество не меняется);
- $sum(l, r)$ — вывести сумму всех элементов x из S, которые удовлетворяют неравенству $l \leqslant x \leqslant r$.

Формат входных данных

Исходно множество S пусто. Первая строка входного файла содержит n — количество операций ($1 \leqslant n \leqslant 300\,000$). Следующие n строк содержат операции. Каждая операция имеет вид либо «`+ i`», либо «`? l r`». Операция «`? l r`» задает запрос $sum(l, r)$.

Если операция «`+ i`» идет во входном файле в начале или после другой операции «`+`», то она задает операцию $add(i)$. Если же она идет после запроса «`?`», и результат этого запроса был y, то выполняется операция $add((i + y) \bmod 10^9)$.

Во всех запросах и операциях добавления параметры лежат в интервале от 0 до 10^9.

Формат выходных данных

Для каждого запроса выведите одно число — ответ на запрос.

Примеры

sum2.in	sum2.out
6 + 1 + 3 + 3 ? 2 4 + 1 ? 2 4	3 7

K-ый максимум

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

Напишите программу, реализующую структуру данных, позволяющую добавлять и удалять элементы, а также находить k-й максимум.

Формат входных данных

Первая строка входного файла содержит натуральное число n — количество команд ($n \leqslant 100\,000$). Последующие n строк содержат по одной команде каждая. Команда записывается в виде двух чисел c_i и k_i — тип и аргумент команды соответственно ($|k_i| \leqslant 10^9$). Поддерживаемые команды:

- $+1$ (или просто 1): Добавить элемент с ключом k_i.
- 0: Найти и вывести k_i-й максимум.
- -1: Удалить элемент с ключом k_i.

Гарантируется, что в процессе работы в структуре не требуется хранить элементы с равными ключами или удалять несуществующие элементы. Также гарантируется, что при запросе k_i-го максимума, он существует.

Формат выходных данных

Для каждой команды нулевого типа в выходной файл должна быть выведена строка, содержащая единственное число — k_i-й максимум.

Примеры

kthmax.in	kthmax.out
11 +1 5 +1 3 +1 7 0 1 0 2 0 3 -1 5 +1 10 0 1 0 2 0 3	7 5 3 10 7 3

Легчайшая

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

Имеется N наборов чисел, которые изначально известны.

Поступают Q запросов вида x y k, для выполнения запроса нужно взять все числа из набора под номером x, которые не меньше числа k, и переместить их в набор номер y. После выполнения всех запросов необходимо вывести конечные состояния наборов.

Формат входных данных

В первой строке дано два числа — N и Q ($2 \leqslant N \leqslant 10^5$, $1 \leqslant Q \leqslant 10^5$).

В последующих N строках заданы наборы. Каждый набор задается строкой вида: число k, за ним k чисел в неубывающем порядке. Суммарный размер наборов не превышает 10^5. Все числа в наборах от 1 до 10^6.

Далее в Q строках заданы запросы — по три числа x, y и k ($1 \leqslant x, y \leqslant N$, $x \neq y$, $1 \leqslant k \leqslant 10^6$).

Формат выходных данных

Выведите N строк — конечные состояния наборов, выводить числа набора следует в неубывающем порядке.

Примеры

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