

## Содержание

<b>Корневая</b>	<b>2</b>
Задача 3А. Фаброзавры-дизайнеры [6 sec, 256 mb]	2
Задача 3В. И снова сумма... [2 sec, 256 mb]	3
Задача 3С. Неявный Ключ [2 sec, 256 mb]	4
<b>Бонусные задачи</b>	<b>5</b>
Задача 3D. Перестановки strike back [5 sec, 256 mb]	5
Задача 3Е. Вставка ключевых значений [5 sec, 256 mb]	6

---

В некоторых задачах большой ввод и вывод. Пользуйтесь **быстрым вводом-выводом**.

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

## Корневая

### Задача 3А. Фаброзавры-дизайнеры [6 сек, 256 mb]

Фаброзавры известны своим тонким художественным вкусом и увлечением ландшафтным дизайном. Они живут около очень живописной реки и то и дело перестраивают тропинку, идущую вдоль реки: либо насыпают дополнительной земли, либо срывают то, что есть. Для того, чтобы упростить эти работы, они поделили всю тропинку на горизонтальные участки, пронумерованные от 1 до  $N$ , и их переделки устроены всегда одинаково: они выбирают часть дороги от  $L$ -ого до  $R$ -ого участка (включительно) и изменяют (увеличивают или уменьшают) высоту на всех этих участках на одну и ту же величину (если до начала переделки высоты были разными, то и после переделки они останутся разными).

Поскольку, как уже говорилось, у фаброзавров тонкий художественный вкус, каждый из них считает, что их река лучше всего выглядит с определенной высоты. Поэтому им хочется знать, есть ли поблизости от их дома место на тропинке, где высота на их взгляд оптимальна. Помогите им в этом разобраться.

#### Формат входных данных

Первая строка входного файла содержит два числа  $N$  и  $M$  — длину дороги и количество запросов соответственно ( $1 \leq N, M \leq 10^5$ ). На второй строке содержатся  $N$  чисел, разделенных пробелами — начальные высоты соответствующих частей дороги; высоты не превосходят  $10^4$  по модулю. В следующих  $M$  строках содержатся запросы по одному на строке.

Запрос  $+ L R X$  означает, что высоту частей дороги от  $L$ -ой до  $R$ -ой (включительно) нужно изменить на  $X$ . При этом  $1 \leq L \leq R \leq N$ , а  $|X| \leq 10^4$ .

Запрос  $? L R X$  означает, что нужно проверить, есть ли между  $L$ -ым и  $R$ -ым участками (включая эти участки) участок, где дорога проходит точно на высоте  $X$ . Гарантируется, что  $1 \leq L \leq R \leq N$ , а  $|X| \leq 10^9$ .

#### Формат выходных данных

На каждый запрос второго типа нужно вывести в выходной файл на отдельной строке одно слово «YES» (без кавычек), если нужный участок существует, и «NO» в противном случае.

#### Примеры

stdin	stdout
10 5	NO
0 1 1 3 3 3 2 0 0 1	YES
? 3 5 2	YES
+ 1 4 1	
? 3 5 2	
+ 7 10 2	
? 9 10 3	

### Задача 3В. И снова сумма... [2 сек, 256 mb]

Реализуйте структуру данных, которая поддерживает множество  $S$  целых чисел, с которым 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  $l \leq x \leq r$ .

#### Формат входных данных

Исходно множество  $S$  пусто. Первая строка входного файла содержит  $n$  — количество операций ( $1 \leq n \leq 300\,000$ ). 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.

#### Пример

stdin	stdout
6	3
+ 1	7
+ 3	
+ 3	
? 2 4	
+ 1	
? 2 4	

#### Замечание

Это упражнение на **корневую!**

### Задача 3С. Неявный Ключ [2 sec, 256 mb]

Научитесь быстро делать две операции с массивом:

- `add i x` — добавить после  $i$ -го элемента  $x$  ( $0 \leq i \leq n$ )
- `del i` — удалить  $i$ -й элемент ( $1 \leq i \leq n$ )

#### Формат входных данных

На первой строке  $n_0$  и  $m$  ( $1 \leq n_0, m \leq 10^5$ ) — длина исходного массива и количество запросов. На второй строке  $n_0$  целых чисел от 0 до  $10^9 - 1$  — исходный массив. Далее  $m$  строк, содержащие запросы. Гарантируется, что запросы корректны: например, если просят удалить  $i$ -й элемент, он точно есть.

#### Формат выходных данных

Выведите конечное состояние массива. На первой строке количество элементов, на второй строке сам массив.

#### Примеры

stdin	stdout
3 4	3
1 2 3	9 2 8
del 3	
add 0 9	
add 3 8	
del 2	

#### Замечание

Это упражнение на **корневую!**

## Бонусные задачи

### Задача 3D. Перестановки strike back [5 sec, 256 mb]

Вася выписал на доске в каком-то порядке все числа от 1 по  $N$ , каждое число ровно по одному разу. Иногда он стирает какое-то число и записывает на его место другое. Количество чисел, выписанных Васей, оказалось довольно большим, поэтому Вася не может окинуть взглядом все числа. Однако ему надо всё-таки представлять эту последовательность, поэтому он написал программу, которая в любой момент 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  $1 \leq M \leq 100\,000$  — суммарное количество вопросов и изменений сделанных Васей. Во второй строке дано  $N$  чисел — последовательность чисел, выписанных Васей. Далее в  $M$  строках находятся описания вопросов. Каждый запрос на изменение числа в некоторой позиции начинается со слова SET и имеет вид SET  $a$   $b$  ( $1 \leq a \leq N$ ,  $1 \leq b \leq N$ ). Это означает, что Вася изменил число, записанное в позиции  $a$  на число  $b$ . Каждый Васин вопрос начинается со слова GET и имеет вид GET  $x$   $y$   $k$   $l$  ( $1 \leq x \leq y \leq N$ ,  $1 \leq k \leq l \leq N$ ).

#### Формат выходных данных

Для каждого Васиного вопроса выведите единственное число — ответ на Васиин вопрос.

#### Пример

stdin	stdout
4 4	1
1 2 3 4	3
GET 1 2 2 3	2
GET 1 3 1 3	
SET 1 4	
GET 1 3 1 3	

#### Замечание

Можно честно запилить 2D-запрос, но корневая может сильно упростить жизнь.

### Задача 3Е. Вставка ключевых значений [5 sec, 256 mb]

Вас наняла на работу компания MacroHard, чтобы вы разработали новую структуру данных для хранения целых ключевых значений.

Эта структура выглядит как массив  $A$  бесконечной длины, ячейки которого нумеруются с единицы. Изначально все ячейки пусты. Единственная операция, которую необходимо поддерживать — это операция  $Insert(L, K)$ , где  $L$  — положение в массиве, а  $K$  — некоторое положительное целое ключевое значение.

Операция выполняется следующим образом:

- Если ячейка  $A[L]$  пуста, то присвоить  $A[L] := K$ .
- Если ячейка  $A[L]$  непуста, выполнить  $Insert(L + 1, A[L])$ , а затем присвоить  $A[L] := K$ .

По заданной последовательности из  $N$  целых чисел  $L_1, L_2, \dots, L_N$  вам необходимо вывести содержимое этого массива после выполнения следующей последовательности операций:

$Insert(L_1, 1)$   
 $Insert(L_2, 2)$   
...  
 $Insert(L_N, N)$

#### Формат входных данных

В первой строке входного файла содержится  $N$  — число операций  $Insert$  и  $M$  — максимальный номер позиции, которую можно использовать в операции  $Insert$ . ( $1 \leq N \leq 131\,072$ ,  $1 \leq M \leq 131\,072$ ).

В следующей строке даны  $N$  целых чисел  $L_i$ , которые описывают операции  $Insert$  ( $1 \leq L_i \leq M$ ).

#### Формат выходных данных

Выведите содержимое массива после выполнения данной последовательности операций  $Insert$ . На первой строке выведите  $W$  — номер последней несвободной позиции в массиве. Далее выведите  $W$  целых чисел —  $A[1], A[2], \dots, A[W]$ . Для пустых ячеек выводите нули.

#### Пример

stdin	stdout
5 4	6
3 3 4 1 3	4 0 5 2 3 1

#### Замечание

Вы же говорили, что знаете декартово дерево?