

## Задача А. Сумма на отрезке

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
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:

- А  $i$   $x$  — 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.

### Пример

стандартный ввод	стандартный вывод
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 В. Фенвик с подвохом

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 0.4 секунд  
Ограничение по памяти: 256 мегабайт

Вам дан массив из  $n$  элементов, изначально заполненный нулями. Вам поступают запросы двух типов:

1. прибавить ко всем числам на  $[L : R]$  число  $x$
2. узнать значение  $i$ -го элемента.

Реализуйте дерево Фенвика, которое поддерживает данные операции. Нет, мы не ошиблись. Да, это дерево Фенвика. Да, оно на это способно (а мы верим, что вы способны это придумать). Нет, слишком долго работать не будет. И да, дерево отрезков здесь не зайдет, даже не пытайтесь.

### Формат входных данных

В первой строке даны два числа  $N$  и  $M$  — количество элементов массива и количество запросов соответственно ( $1 \leq N \leq 2 \cdot 10^5, 1 \leq M \leq 6 \cdot 10^5$ )

В следующих  $m$  строках описаны запросы:

Запросы первого типа имеют вид  $+LRx$  ( $1 \leq L, R \leq N, -100 \leq x \leq 100$ ).

Запросы второго типа имеют вид  $?i$  ( $1 \leq i \leq N$ )

### Формат выходных данных

На каждый запрос второго типа выведите элемент стоящий на  $i$ -й позиции массива.

### Пример

стандартный ввод	стандартный вывод
5 5	7
+ 1 5 7	10
? 3	7
+ 1 3 3	
? 2	
? 4	

## Задача С. Звезды

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
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.

### Пример

стандартный ввод	стандартный вывод
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 D. Максимум на подотрезках с добавлением на отрезке

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 0.5 секунд  
Ограничение по памяти: 256 мегабайт

Реализуйте эффективную структуру данных для хранения массива и выполнения следующих операций: увеличение всех элементов данного интервала на одно и то же число; поиск максимума на интервале.

### Формат входных данных

В первой строке вводится одно натуральное число  $N$  ( $1 \leq N \leq 100000$ ) — количество чисел в массиве.

Во второй строке вводятся  $N$  чисел от 0 до 100000 — элементы массива.

В третьей строке вводится одно натуральное число  $M$  ( $1 \leq M \leq 30000$ ) — количество запросов.

Каждая из следующих  $M$  строк представляет собой описание запроса. Сначала вводится одна буква, кодирующая вид запроса ( $m$  — найти максимум,  $a$  — увеличить все элементы на отрезке).

Следом за  $m$  вводятся два числа — левая и правая граница интервала.

Следом за  $a$  вводятся три числа — левый и правый концы отрезка и число  $add$ , на которое нужно увеличить все элементы данного отрезка массива ( $0 \leq add \leq 100000$ ).

### Формат выходных данных

Выведите в одну строку через пробел ответы на каждый запрос  $m$ .

### Пример

стандартный ввод	стандартный вывод
5	4 104 104
2 4 3 1 5	
5	
m 1 3	
a 2 4 100	
m 1 3	
a 5 5 10	
m 1 5	

## Задача E. XOR на отрезке

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 3 секунды  
Ограничение по памяти: 512 мегабайт

Вам задан массив  $a$ , состоящий из  $n$  целых чисел  $a_1, a_2, \dots, a_n$ . С этим массивом разрешается выполнять две операции:

1. Вычислить сумму текущих элементов массива на отрезке  $[l, r]$ , то есть посчитать значение  $a_l + a_{l+1} + \dots + a_r$
2. Применить операцию хог с заданным числом  $x$  к каждому элементу массива на отрезке  $[l, r]$ , то есть выполнить  $a_l = a_l \oplus x, a_{l+1} = a_{l+1} \oplus x, \dots, a_r = a_r \oplus x$ . Эта операция изменяет ровно  $r - l + 1$  элементов массива.

Выражение  $x \oplus y$  означает применение побитовой операции хог к числам  $x$  и  $y$ .

Вам задан список из  $m$  операций указанного вида. От Вас требуется выполнить все заданные операции, для каждого запроса суммы требуется вывести полученный результат.

### Формат входных данных

Входные данные В первой строке задано целое число  $n$  ( $1 \leq n \leq 10^5$ ) - размер массива. Во второй строке через пробел заданы целые числа  $a_1, a_2, \dots, a_n$  ( $0 \leq a_i \leq 10^6$ ) - исходный массив.

В третьей строке задано целое число  $m$  ( $1 \leq m \leq 10^5$ ) - количество операций с массивом. В  $i$ -ой из следующих  $m$  строк сперва записано целое число  $t_i$  ( $1 \leq t_i \leq 2$ ) - тип  $i$ -го запроса. Если  $t_i = 1$ , то это запрос суммы, если  $t_i = 2$ , то это запрос на изменение элементов массива. Если  $i$ -ая операция типа 1, то далее следуют два целых числа  $l_i, r_i$  ( $1 \leq l_i \leq r_i \leq n$ ). Если  $i$ -ая операция типа 2, то далее следуют три целых числа  $l_i, r_i, x_i$  ( $1 \leq l_i \leq r_i \leq n, 1 \leq x_i \leq 10^6$ ). Числа в строках разделены одиночными пробелами.

### Формат выходных данных

Для каждого запроса типа 1 в отдельной строке выведите сумму чисел на требуемом отрезке. Ответы на запросы выводите в том порядке, в котором они заданы во входных данных.

### Примеры

стандартный ввод	стандартный вывод
5 4 10 3 13 7 8 1 2 4 2 1 3 3 1 2 4 1 3 3 2 2 5 5 1 1 5 2 1 2 10 1 2 3	26 22 0 34 11
6 4 7 4 0 7 3 5 2 2 3 8 1 1 5 2 3 5 1 2 4 5 6 1 2 3	38 28

## Задача F. Простое задание

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 512 мегабайт

Это задание очень простое. Вам дана строка  $S$  длины  $n$  и  $q$  запросов, каждый запрос имеет формат  $i j k$ , что означает: отсортировать подстроку, состоящую из символов от  $i$  до  $j$ , в неубывающем порядке, если  $k = 1$  или в невозрастающем порядке, если  $k = 0$ .

Выведите итоговую строку после выполнения запросов.

### Формат входных данных

В первой строке записано два целых числа  $n, q$  ( $1 \leq n \leq 10^5, 0 \leq q \leq 50000$ ), длина строки и количество запросов, соответственно.

В следующей строке идёт сама строка  $S$ . Она состоит только из строчных английских букв.

В каждой из следующих  $q$  строк записано по три целых числа  $i, j, k$  ( $1 \leq i \leq j \leq n, 0 \leq k \leq 1$ ), обозначающих запрос.

### Формат выходных данных

Выведите строку  $S$  после выполнения всех запросов.

### Пример

стандартный ввод	стандартный вывод
10 10 ittmcsvmoa 6 7 0 2 4 0 4 10 1 1 2 0 2 9 1 5 6 1 7 9 0 1 2 0 2 6 0 3 9 1	tmacimostv

## Задача G. Присваивание на отрезке

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 0.5 секунд  
Ограничение по памяти: 256 мегабайт

Реализуйте эффективную структуру данных для хранения элементов и присваивания нескольким подряд идущим элементам одного и того же числа.

### Формат входных данных

В первой строке вводится одно натуральное число  $N$  ( $1 \leq N \leq 100000$ ) — количество чисел в массиве.

Во второй строке вводятся  $N$  чисел от 0 до 100000 — элементы массива.

В третьей строке вводится одно натуральное число  $M$  ( $1 \leq M \leq 30000$ ) — количество запросов.

Каждая из следующих  $M$  строк представляет собой описание запроса. Сначала вводится одна буква, кодирующая вид запроса ( $g$  — получить текущее значение элемента по его номеру,  $a$  — присвоить всем элементам отрезка новое значение).

Следом за  $g$  вводится одно число — номер элемента.

Следом за  $a$  вводятся три числа — левый и правый концы отрезка и число  $value$ , которое нужно присвоить всем элементам данного отрезка массива ( $0 \leq value \leq 100000$ ).

### Формат выходных данных

Выведите в одну строку через пробел ответы на каждый запрос  $g$ .

### Пример

стандартный ввод	стандартный вывод
5	3
2 4 3 1 5	10
4	2
g 3	
a 2 4 10	
g 3	
g 1	

## Задача Н. Присваивание, прибавление и сумма

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 1024 мегабайта

Есть массив из  $n$  элементов, изначально заполненный нулями. Вам нужно написать структуру данных, которая обрабатывает три вида запросов:

- присвоить всем элементам на отрезке от  $l$  до  $r - 1$  значение  $v$ ,
- прибавить ко всем элементам на отрезке от  $l$  до  $r - 1$  число  $v$ ,
- узнать сумму на отрезке от  $l$  до  $r - 1$ .

### Формат входных данных

Первая строка содержит два числа  $n$  и  $m$  ( $1 \leq n, m \leq 100000$ ) — размер массива и число операций. Далее следует описание операций. Описание каждой операции имеет следующий вид:

- $1\ l\ r\ v$  — присвоить всем элементам на отрезке от  $l$  до  $r - 1$  значение  $v$  ( $0 \leq l < r \leq n$ ,  $0 \leq v \leq 10^5$ ).
- $2\ l\ r\ v$  — прибавить ко всем элементам на отрезке от  $l$  до  $r - 1$  число  $v$  ( $0 \leq l < r \leq n$ ,  $0 \leq v \leq 10^5$ ).
- $3\ l\ r$  — узнать сумму на отрезке от  $l$  до  $r - 1$  ( $0 \leq l < r \leq n$ ).

### Формат выходных данных

Для каждой операции третьего типа выведите соответствующее значение.

### Примеры

стандартный ввод	стандартный вывод
5 7	8
1 0 3 3	10
2 2 4 2	4
3 1 3	
2 1 5 1	
1 0 2 2	
3 0 3	
3 3 5	



## Задача I. RMQ наоборот

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	3 секунды
Ограничение по памяти:	64 мегабайта

Рассмотрим массив  $a[1..n]$ . Пусть  $Q(i, j)$  — ответ на запрос о нахождении минимума среди чисел  $a[i], \dots, a[j]$ . Вам даны несколько запросов и ответы на них. Восстановите исходный массив.

### Формат входных данных

Первая строка входного файла содержит число  $n$  — размер массива, и  $m$  — число запросов ( $1 \leq n, m \leq 100\,000$ ). Следующие  $m$  строк содержат по три целых числа  $i, j$  и  $q$ , означающих, что  $Q(i, j) = q$  ( $1 \leq i \leq j \leq n, -2^{31} \leq q \leq 2^{31} - 1$ ).

### Формат выходных данных

Если искомого массива не существует, выведите строку «**inconsistent**».

В противном случае в первую строку выходного файла выведите «**consistent**». Во вторую строку выходного файла выведите элементы массива. Элементами массива должны быть целые числа в интервале от  $-2^{31}$  до  $2^{31} - 1$  включительно. Если решений несколько, выведите любое.

### Примеры

стандартный ввод	стандартный вывод
3 2 1 2 1 2 3 2	consistent 1 2 2
3 3 1 2 1 1 1 2 2 3 2	inconsistent

## Задача J. Блин, че-то я тупанул, конечно

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Письмо на Балабановскую спичечную фабрику: «Я 11 лет считаю спички у вас в коробках — их то 59, то 60, а иногда и 58. Вы там сумасшедшие что ли все???»

Гриша очень педантичный препод, который любит спорт. Поэтому он записал количество голов, которые он, *по его словам*, забил за каждый футбольный матч (может даже отрицательное, если он пропускал голы).

Еще Гриша считает, что главное в спорте — прогресс. Но прогресс должен быть плавным. Поэтому он хочет для разных последовательностей дней узнавать, насколько же прогресс был плавным. Мера плавности — длина максимального подотрезка вида  $(x, x + 1, \dots, x + l)$ , где  $l$  — длина такого отрезка.

Сокомандник Гриши, Женя, завидует результативности Гриши. Поэтому иногда он убавляет количество голов в блокнотике Гриши за некоторые матчи (иногда даже добавляет, чтобы ему не так обидно было).

А еще Гриша старший преподаватель, поэтому у него нет времени че-то там считать, поэтому вам дали данную задачу.

### Формат входных данных

В первой строке находится одно целое число  $n$  — количество матчей по футболу за смену, в которых Гриша принял участие.  $1 \leq n \leq 10^5$  Во второй строке находятся  $n$  чисел  $q_i$  ( $|q_i| \leq 10^9$ ) — количество голов, который Гриша забил в каждом матче. В третьей строке находится одно целое число  $m$  ( $0 \leq m \leq 100\,000$ ) — количество действий с блокнотиком Гриши. В следующих  $m$  строках содержится описание эксперимента.

- $+ l_i r_i d_i$  — Женя добавил  $d_i$  голов в матчах с  $l_i$ -го по  $r_i$ -й. ( $1 \leq l_i \leq r_i \leq n$ ,  $|d_i| \leq 10^9$ )
- $? l_i r_i$  — Гриша попросил Вас узнать насколько у него был плавный прогресс в матчах с  $l_i$ -го по  $r_i$ -й. ( $1 \leq l_i \leq r_i \leq n$ )

### Формат выходных данных

Для каждого действия второго типа выведите в новой строке одно число — меру плавности.

### Пример

стандартный ввод	стандартный вывод
6	
2 3 4 3 4 4	
5	
? 1 6	
+ 6 6 1	
? 2 6	
+ 4 6 2	
? 1 5	