

Задача А. Снеговики

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

Зима. 2012 год. На фоне грядущего Апокалипсиса и конца света незамеченной прошла новость об очередном прорыве в областях клонирования и снеговиков: клонирования снеговиков. Вы конечно знаете, но мы вам напомним, что снеговик состоит из нуля или более вертикально поставленных друг на друга шаров, а клонирование — это процесс создания идентичной копии (клона).

В местечке Местячково учитель Андрей Сергеевич Учитель купил через интернет-магазин «Интернет-магазин аппаратов клонирования» аппарат для клонирования снеговиков. Теперь дети могут играть и даже играют во дворе в следующую игру. Время от времени один из них выбирает понравившегося снеговика, клонирует его и:

- либо добавляет ему сверху один шар;
- либо удаляет из него верхний шар (если снеговик не пустой).

Учитель Андрей Сергеевич записал последовательность действий и теперь хочет узнать суммарную массу всех построенных снеговиков.

Формат входных данных

Первая строка содержит количество действий n ($1 \leq n \leq 200\,000$). В строке номер $i + 1$ содержится описание действия i :

- $t\ m$ — клонировать снеговика номер t ($0 \leq t < i$) и добавить сверху шар массой m ($0 < m \leq 1000$);
- $t\ 0$ — клонировать снеговика номер t ($0 \leq t < i$) и удалить верхний шар. Гарантируется, что снеговик t не пустой.

В результате действия i , описанного в строке $i + 1$ создается снеговик номер i . Изначально имеется пустой снеговик с номером ноль.

Все числа во входном файле целые.

Формат выходных данных

Выведите суммарную массу построенных снеговиков.

Примеры

snowmen.in	snowmen.out
8	74
0 1	
1 5	
2 4	
3 2	
4 3	
5 0	
6 6	
1 0	

Задача В. НВП на дереве

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

Дано дерево на N вершинах, у которого i -е ребро соединяет вершину u_i и вершину v_i . На вершине с номером i записано целое число a_i . Для каждого целого числа k от 1 до N решите следующую задачу:

Составим последовательность, выписав целые числа на вершинах, вдоль кратчайшего пути от вершины 1 до вершины k , в том порядке, в котором они появляются. Найдите длину наибольшей возрастающей подпоследовательности этой последовательности.

Формат входных данных

В первой строке вводится число n ($2 \leq n \leq 2 \cdot 10^5$) — количество вершин в дереве.

Во второй строке через пробел задаются числа a_i ($1 \leq a_i \leq 10^9$) — числа, записанные на вершинах.

В каждой из следующих $n - 1$ -й строке вводятся пары v_i, u_i — рёбра дерева ($1 \leq v_i \neq u_i \leq n$). Гарантируется, что данный набор рёбер образует дерево.

Формат выходных данных

Выведите N строк. В k -й строке выведите длину наибольшей возрастающей подпоследовательности последовательности, полученной по кратчайшему пути из вершины 1 в вершину k .

Примеры

стандартный ввод	стандартный вывод
10	1
1 2 5 3 4 6 7 3 2 4	2
1 2	3
2 3	3
3 4	4
4 5	4
3 6	5
6 7	2
1 8	2
8 9	3
9 10	

Задача С. K -я порядковая статистика на отрезке

Имя входного файла: `kth.in`
Имя выходного файла: `kth.out`
Ограничение по времени: 6 секунд
Ограничение по памяти: 512 мегабайт

Дан массив из N неотрицательных чисел, строго меньших 10^9 . Вам необходимо ответить на несколько запросов о величине k -й порядковой статистики на отрезке $[l, r]$.

Формат входных данных

Первая строка содержит число N ($1 \leq N \leq 450\,000$) — размер массива.

Вторая строка может быть использована для генерации a_i — начальных значений элементов массива. Она содержит три числа a_1 , l и m ($0 \leq a_1, l, m < 10^9$); для i от 2 до N

$$a_i = (a_{i-1} \cdot l + m) \bmod 10^9.$$

В частности, $0 \leq a_i < 10^9$.

Третья строка содержит одно целое число B ($1 \leq B \leq 1000$) — количество групп запросов.

Следующие B строк описывают одну группу запросов. Каждая группа запросов описывается 10 числами. Первое число G обозначает количество запросов в группе. Далее следуют числа x_1 , l_x и m_x , затем y_1 , l_y и m_y , затем, k_1 , l_k и m_k ($1 \leq x_1 \leq y_1 \leq N$, $1 \leq k_1 \leq y_1 - x_1 + 1$, $0 \leq l_x, m_x, l_y, m_y, l_k, m_k < 10^9$). Эти числа используются для генерации вспомогательных последовательностей x_g и y_g , а также параметров запросов i_g , j_g и k_g ($1 \leq g \leq G$)

$$\begin{aligned}x_g &= ((i_{g-1} - 1) \cdot l_x + m_x) \bmod N + 1, & 2 \leq g \leq G \\y_g &= ((j_{g-1} - 1) \cdot l_y + m_y) \bmod N + 1, & 2 \leq g \leq G \\i_g &= \min(x_g, y_g), & 1 \leq g \leq G \\j_g &= \max(x_g, y_g), & 1 \leq g \leq G \\k_g &= (((k_{g-1} - 1) \cdot l_k + m_k) \bmod (j_g - i_g + 1)) + 1, & 2 \leq g \leq G\end{aligned}$$

Сгенерированные последовательности описывают запросы, g -й запрос состоит в поиске k_g -го по величине числа среди элементов отрезка $[i_g, j_g]$.

Суммарное количество запросов не превосходит 600 000.

Формат выходных данных

Выведите единственное число — сумму ответов на запросы.

Примеры

kth.in	kth.out
5 1 1 1 5 1 1 0 0 3 0 0 2 0 0 1 2 0 0 5 0 0 3 0 0 1 1 0 0 5 0 0 5 0 0 1 3 0 0 3 0 0 1 0 0 1 1 0 0 4 0 0 1 0 0	15

Задача D. Менеджер памяти

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 8 секунды
Ограничение по памяти: 512 мегабайт

Одно из главных нововведений новейшей операционной системы Indows 7 — новый менеджер памяти. Он работает с массивом длины N и позволяет выполнять три самые современные операции:

- `copy(a, b, l)` — скопировать отрезок длины $[a, a + l - 1]$ в $[b, b + l - 1]$
- `sum(l, r)` — посчитать сумму элементов массива на отрезке $[l, r]$
- `print(l, r)` — напечатать элементы с l по r , включительно

Вы являетесь разработчиком своей операционной системы, и Вы, безусловно, не можете обойтись без инновационных технологий. Вам необходимо реализовать точно такой же менеджер памяти.

Формат входных данных

Первая строка входного файла содержит целое число N ($1 \leq N \leq 1\,000\,000$) — размер массива, с которым будет работать Ваш менеджер памяти.

Во второй строке содержатся четыре числа $1 \leq X_1, A, B, M \leq 10^9 + 10$. С помощью них можно сгенерировать исходный массив чисел $X_1, X_2, \dots, X_N$. $X_{i+1} = (A * X_i + B) \bmod M$

Следующая строка входного файла содержит целое число K ($1 \leq K \leq 200\,000$) — количество запросов, которые необходимо выполнить Вашему менеджеру памяти.

Далее в K строках содержится описание запросов. Запросы заданы в формате:

- `cpy a b l` — для операции `copy`
- `sum l r` — для операции `sum` ($l \leq r$)
- `out l r` — для операции `print` ($l \leq r$)

Гарантируется, что суммарная длина запросов `print` не превышает 3 000. Также гарантируется, что все запросы корректны.

Формат выходных данных

Для каждого запроса `sum` или `print` выведите в выходной файл на отдельной строке результат запроса.

Примеры

stdin	stdout
6	1 2 6 1 2 6
1 4 5 7	1 2 1 2 2 6
7	6
out 1 6	1 1 2 1 2 6
cpy 1 3 2	13
out 1 6	
sum 1 4	
cpy 1 2 4	
out 1 6	
sum 1 6	

Задача Е. Испытание силомера

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	3 секунды
Ограничение по памяти:	256 мегабайт

Сайтама выполняет последовательные удары по силомеру. Силомер представляет из себя массив целых чисел длины n . Изначально i -е число массива равно a_i для всех i .

Вам необходимо обработать q событий, происходящих с силомером. Событие номер i может быть одного из трех типов:

1. подходит наблюдатель и просит посчитать сумму чисел массива на отрезке $[l_i; r_i]$, то есть величину $a_{l_i} + a_{l_i+1} + \dots + a_{r_i}$;
2. Сайтама наносит обычный удар силы x_i по отрезку $[l_i; r_i]$: всем элементам массива на позициях от l_i до r_i включительно присваивается значение x_i
3. Сайтама наносит сильный удар по отрезку $[l_i; r_i]$: для всех j от l_i до r_i включительно происходит присваивание $a_j \leftarrow \text{popcount}(a_j)$.

Здесь $\text{popcount}(x)$ — это количество единичных бит в двоичной записи числа x . Иными словами, при событии третьего типа каждое число на отрезке события заменяется на количество своих единичных бит.

На каждый подход наблюдателя, то есть событие первого типа, сообщите ему интересующую его сумму.

Формат входных данных

В первой строке записаны два целых числа n и q — длина массива и количество событий ($1 \leq n, q \leq 2 \cdot 10^5$).

Во второй строке через пробел записаны n целых чисел $a_1, \dots, a_n$ — начальные элементы массива силомера ($0 \leq a_i \leq 10^9$).

Следующие q строк описывают события. Первое число t_i в описании события — тип события ($1 \leq t \leq 3$). Следующие два заданные через пробел числа — это границы отрезка l_i и r_i ($1 \leq l_i \leq r_i \leq n$). Если это событие второго типа, то есть $t_i = 2$, далее следует число x_i , обозначающее, что надо выполнить присваивания $a_j \leftarrow x_i$ для всех $l_i \leq j \leq r_i$ ($0 \leq x_i \leq 10^9$).

Формат выходных данных

Для каждого события первого типа выведите в отдельной строке сумму элементов массива на отрезке, заданном этим событием.

Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты этой подзадачи и необходимых подзадач, а также тесты из условия успешно пройдены.

Подзадача	Баллы	Дополнительные ограничения	Необходимые подзадачи	Информация о проверке
1	12	$n, q \leq 5000$	—	полная
2	7	все a_i и x_i — степени двойки	—	полная
3	4	все a_i и x_i — степени двойки или нули	2	полная
4	9	$t_i \neq 2$ для всех i	—	полная
5	5	для всех событий второго типа $l_i = r_i$	4	полная
6	12	$a_i, x_i \leq 20$	—	полная
7	15	$n \leq 10^5$ и $q \leq 10^4$	1	полная
8	16	все события первого типа следуют строго позже событий второго и третьего типов	—	полная
9	20	нет	1 – 8	первая ошибка

Примеры

стандартный ввод	стандартный вывод
6 6 9 1 3 1 6 3 2 2 3 9 2 4 5 10 3 2 4 1 2 4 1 2 6 3 2 5	6 19
5 6 0 3 4 1 1 1 2 5 2 2 3 8 3 1 4 3 2 2 1 3 4 2 3 5 5	9 2

Задача F. Исторический максимум

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

Дан массив a . Поступают запросы прибавления на отрезке и поиска максимума исторических максимумов на отрезке. Необходимо их обрабатывать.

Формат входных данных

В первой строке дано число n ($1 \leq n \leq 300\,000$) — количество элементов массива a .

Во второй строке даны n чисел — элементы массива a ($-10^9 \leq a_i \leq 10^9$).

В третьей строке дано число q ($1 \leq q \leq 300\,000$) — количество запросов.

В последующих q строках даны запросы.

Запрос прибавления на отрезке задается следующим образом: 1 ql qr x

Это означает, что на отрезке от ql до qr включительно ($1 \leq ql \leq qr \leq n$) нужно прибавить ко всем числам x ($-10^9 \leq x \leq 10^9$).

Запрос поиска исторического максимума задается следующим образом: 2 ql qr

Это означает, что нужно найти максимум исторических максимумов на отрезке от ql до qr включительно ($1 \leq ql \leq qr \leq n$).

Формат выходных данных

Для каждого запроса второго типа выведите ответ в отдельной строке.

Пример

стандартный ввод	стандартный вывод
5	6
1 2 3 4 5	6
5	6
1 1 4 2	
2 2 5	
1 3 5 -5	
2 4 5	
2 1 5	

Задача G. Дели, прибавляй!

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 4 секунды
Ограничение по памяти: 256 мегабайт

Дан массив целых чисел a длины n . Поступает q запросов четырех типов:

- $1\ l\ r\ x$. Для каждого i на отрезке от l до r включительно нужно заменить a_i на $a_i + x$.
- $2\ l\ r\ x$. Для каждого i на отрезке от l до r включительно нужно заменить a_i на $\lfloor \frac{a_i}{x} \rfloor$ ($\lfloor \cdot \rfloor$ — это округление вниз).
- $3\ l\ r$. Необходимо вывести минимум элементов массива a на отрезке от l до r включительно.
- $4\ l\ r$. Необходимо вывести сумму элементов массива a на отрезке от l до r включительно.

Формат входных данных

В первой строке входных данных даны два целых числа n и q ($1 \leq n, q \leq 200\,000$) — количество элементов массива a и количество запросов.

Во второй строке даны n целых чисел a_i ($-10^9 \leq a_i \leq 10^9$) — элементы массива a .

В последующих q строках даны запросы по одному в строке.

Запрос первого типа задается так: $1\ l\ r\ x$

Где $0 \leq l \leq r < n$, $-10^7 \leq x \leq 10^7$ — целые числа. Это означает, что ко всем элементам массива a на отрезке от l до r нужно прибавить x .

Запрос второго типа задается так: $2\ l\ r\ x$

Где $0 \leq l \leq r < n$, $1 \leq x \leq 10^9$ — целые числа. Это означает, что все элементы массива a на отрезке от l до r нужно поделить на x и округлить вниз.

Запрос третьего типа задается так: $3\ l\ r$

Где $0 \leq l \leq r < n$ — целые числа. Это означает, что нужно вывести минимум элементов массива a на отрезке от l до r .

Запрос четвертого типа задается так: $4\ l\ r$

Где $0 \leq l \leq r < n$ — целые числа. Это означает, что нужно вывести сумму элементов массива a на отрезке от l до r .

Формат выходных данных

Для каждого запроса 3 и 4 типов выведите в отдельной строке ответ.

Пример

стандартный ввод	стандартный вывод
7 7	-10
10 -6 4 7 12 1 0	-10
2 1 4 4	-34
1 0 5 -8	-36
3 0 6	-26
3 1 5	
4 0 6	
4 1 5	
4 2 6	

Замечание

Обратите внимание на то, что элементы массива нумеруются с нуля.

Задача Н. Минимизируй!

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

Дан массив целых чисел a длины n . Поступает q запросов двух типов:

- 1 $l\ r\ x$. Для каждого i на отрезке от l до r включительно нужно заменить a_i на $\min(a_i, x)$.
- 2 $l\ r$. Необходимо вывести сумму элементов массива a на отрезке от l до r включительно.

Формат входных данных

В первой строке входных данных дано целое число n ($1 \leq n \leq 300\,000$) — количество элементов массива a .

Во второй строке даны n целых чисел a_i ($1 \leq a_i \leq 10^9$) — элементы массива a .

В третьей строке дано целое число q ($1 \leq q \leq 300\,000$) — количество запросов.

В последующих q строках даны запросы по одному в строке.

Запрос первого типа задается так: 1 $l\ r\ x$

Где $1 \leq l \leq r \leq n$, $1 \leq x \leq 10^9$ — целые числа. Это означает, что все элементы массива a на отрезке от l до r нужно заменить на минимум из текущего значения и x .

Запрос второго типа задается так: 2 $l\ r$

Где $1 \leq l \leq r \leq n$ — целые числа. Это означает, что нужно вывести сумму элементов массива a на отрезке от l до r .

Формат выходных данных

Для каждого запроса второго типа выведите в отдельной строке сумму элементов на соответствующем отрезке.

Примеры

стандартный ввод	стандартный вывод
3 1 4 2 5 2 1 3 1 1 3 3 2 1 3 1 1 3 1 2 1 3	7 6 3
7 1 7 2 4 8 4 100 7 1 3 6 3 2 2 7 1 2 3 5 2 1 7 1 1 7 3 2 1 4 2 2 7	118 117 9 17

Задача I. Минимизируй, прибавляй!

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

Дан массив целых чисел a длины n . Поступает q запросов трех типов:

- $1\ l\ r\ x$. Для каждого i на отрезке от l до r включительно нужно заменить a_i на $\min(a_i, x)$.
- $2\ l\ r\ x$. Для каждого i на отрезке от l до r включительно нужно заменить a_i на $a_i + x$.
- $3\ l\ r$. Необходимо вывести сумму элементов массива a на отрезке от l до r включительно.

Формат входных данных

В первой строке входных данных дано целое число n ($1 \leq n \leq 300\,000$) — количество элементов массива a .

Во второй строке даны n целых чисел a_i ($-10^9 \leq a_i \leq 10^9$) — элементы массива a .

В третьей строке дано целое число q ($1 \leq q \leq 300\,000$) — количество запросов.

В последующих q строках даны запросы по одному в строке.

Запрос первого типа задается так: $1\ l\ r\ x$

Где $1 \leq l \leq r \leq n$, $-10^9 \leq x \leq 10^9$. Это означает, что все элементы массива a на отрезке от l до r нужно заменить на минимум из текущего значения и x .

Запрос второго типа задается так: $2\ l\ r\ x$

Где $1 \leq l \leq r \leq n$, $-10^7 \leq x \leq 10^7$. Это означает, что ко всем элементам массива a на отрезке от l до r нужно прибавить x .

Запрос третьего типа задается так: $3\ l\ r$

Где $1 \leq l \leq r \leq n$. Это означает, что нужно вывести сумму элементов массива a на отрезке от l до r .

Формат выходных данных

Для каждого запроса третьего типа выведите в отдельной строке сумму элементов на соответствующем отрезке.

Примеры

стандартный ввод	стандартный вывод
3 1 4 2 9 3 1 3 1 1 3 3 3 1 3 1 1 3 1 3 1 3 2 1 3 5 3 1 3 1 1 3 3 3 1 3	7 6 3 18 9
7 1 7 2 4 8 4 100 10 1 3 6 3 3 2 7 1 2 3 5 2 3 4 -10 3 1 7 1 1 7 3 3 1 4 3 2 7 2 1 7 5 3 1 7	118 97 -11 -3 33

Задача J. Минимизируй, прибавляй, НОДируй и их друзья

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	4 секунды
Ограничение по памяти:	512 мегабайт

Дан массив целых чисел a длины n . Поступает q запросов восьми типов:

- $1\ l\ r\ x$. Для каждого i на отрезке от l до r включительно нужно заменить a_i на $\min(a_i, x)$.
- $2\ l\ r\ x$. Для каждого i на отрезке от l до r включительно нужно заменить a_i на $\max(a_i, x)$.
- $3\ l\ r\ x$. Для каждого i на отрезке от l до r включительно нужно заменить a_i на x .
- $4\ l\ r\ x$. Для каждого i на отрезке от l до r включительно нужно заменить a_i на $a_i + x$.
- $5\ l\ r$. Необходимо вывести сумму элементов массива a на отрезке от l до r включительно.
- $6\ l\ r$. Необходимо вывести минимум элементов массива a на отрезке от l до r включительно.
- $7\ l\ r$. Необходимо вывести максимум элементов массива a на отрезке от l до r включительно.
- $8\ l\ r$. Необходимо вывести НОД (наибольший общий делитель) элементов массива a на отрезке от l до r включительно.

Формат входных данных

В первой строке входных данных дано целое число n ($1 \leq n \leq 200\,000$) — количество элементов массива a .

Во второй строке даны n целых чисел a_i ($1 \leq a_i \leq 10^9$) — элементы массива a .

В третьей строке дано целое число q ($1 \leq q \leq 200\,000$) — количество запросов.

В последующих q строках даны запросы по одному в строке.

Запрос первого типа задается так: $1\ l\ r\ x$

Где $1 \leq l \leq r \leq n$, $1 \leq x \leq 10^9$ — целые числа. Это означает, что все элементы массива a на отрезке от l до r нужно заменить на минимум из текущего значения и x .

Запрос второго типа задается так: $2\ l\ r\ x$

Где $1 \leq l \leq r \leq n$, $1 \leq x \leq 10^9$ — целые числа. Это означает, что все элементы массива a на отрезке от l до r нужно заменить на максимум из текущего значения и x .

Запрос третьего типа задается так: $3\ l\ r\ x$

Где $1 \leq l \leq r \leq n$, $1 \leq x \leq 10^9$ — целые числа. Это означает, что все элементы массива a на отрезке от l до r нужно заменить на x .

Запрос четвертого типа задается так: $4\ l\ r\ x$

Где $1 \leq l \leq r \leq n$, $1 \leq x \leq 10^7$ — целые числа. Это означает, что ко всем элементам массива a на отрезке от l до r нужно прибавить x .

Запрос пятого типа задается так: $5\ l\ r$

Где $1 \leq l \leq r \leq n$ — целые числа. Это означает, что нужно вывести сумму элементов массива a на отрезке от l до r .

Запрос шестого типа задается так: $6\ l\ r$

Где $1 \leq l \leq r \leq n$ — целые числа. Это означает, что нужно вывести минимум элементов массива a на отрезке от l до r .

Запрос седьмого типа задается так: $7\ l\ r$

Где $1 \leq l \leq r \leq n$ — целые числа. Это означает, что нужно вывести максимум элементов массива a на отрезке от l до r .

Запрос восьмого типа задается так: $8\ l\ r$

Где $1 \leq l \leq r \leq n$ — целые числа. Это означает, что нужно вывести наибольший общий делитель элементов массива a на отрезке от l до r .

Формат выходных данных

Для каждого запроса 5, 6, 7 и 8 типов выведите в отдельной строке ответ.

Пример

стандартный ввод	стандартный вывод
7	71
1 2 3 4 5 6 7	1
20	16
4 2 7 10	1
5 1 6	90
6 1 6	14
7 1 6	17
8 1 6	1
2 1 6 14	74
5 2 7	12
6 2 7	14
7 2 7	2
8 2 7	101
1 2 7 12	12
5 1 6	15
6 1 6	1
7 1 6	
8 1 6	
3 2 6 15	
5 1 7	
6 1 7	
7 1 7	
8 1 7	

Задача К. Ребенок и последовательность

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

В день детей ребенок пришел домой к Пиксу и все перевернул вверх дном. Пикс на него разозлился. В бардаке потерялось много всего, включая любимую последовательность Пикса.

К счастью, Пикс помнит, как можно восстановить последовательность. Сначала нужно завести целочисленный массив $a[1], a[2], \dots, a[n]$. Затем нужно выполнить последовательно m операций. Операции могут быть такими:

1. Операция вывода суммы (параметры l, r). Пикс должен записать значение $\sum_{i=l}^r a[i]$.
2. Операция взятия по модулю (параметры l, r, x). Пикс должен выполнить присвоения $a[i] = a[i] \bmod x$ для каждого i ($l \leq i \leq r$).
3. Операция изменения значения (параметры k, x). Пикс должен изменить значение $a[k]$ на x (иными словами, выполнить присвоение $a[k] = x$).

Сможете ли вы помочь Пиксу выполнить заданную последовательность операций?

Формат входных данных

В первой строке записано два целых числа: n, m ($1 \leq n, m \leq 10^5$). Во второй строке записано n целых чисел через пробел: $a[1], a[2], \dots, a[n]$ ($1 \leq a[i] \leq 10^9$) — начальное значение элементов массива.

Каждая из следующих m строк начинается с целого числа $type$ ($type \in \{1, 2, 3\}$).

- Если $type = 1$, то далее в строке идут два целых числа: l, r ($1 \leq l \leq r \leq n$) — описание операции 1.
- Если $type = 2$, то далее в строке идут еще три целых числа: l, r, x ($1 \leq l \leq r \leq n$; $1 \leq x \leq 10^9$) — описание операции 2.
- Если $type = 3$, то далее в строке идут два целых числа: k, x ($1 \leq k \leq n$; $1 \leq x \leq 10^9$) — описание операции 3.

Формат выходных данных

Для каждой операции 1, выведите значение, которое должен записать Пикс. Обратите внимание, что ответ может не помещаться в 32-битное целое число.

Примеры

стандартный ввод	стандартный вывод
5 5 1 2 3 4 5 2 3 5 4 3 3 5 1 2 5 2 1 3 3 1 1 3	8 5
10 10 6 9 6 7 6 1 10 10 9 5 1 3 9 2 7 10 9 2 5 10 8 1 4 7 3 3 7 2 7 9 9 1 2 4 1 6 6 1 5 9 3 1 10	49 15 23 1 9

Замечание

Рассмотрим первый тестовый пример:

- Сперва $a = \{1, 2, 3, 4, 5\}$.
- После операции 1, $a = \{1, 2, 3, 0, 1\}$.
- После операции 2, $a = \{1, 2, 5, 0, 1\}$.
- При операции 3, $2 + 5 + 0 + 1 = 8$.
- После операции 4, $a = \{1, 2, 2, 0, 1\}$.
- При операции 5, $1 + 2 + 2 = 5$.

Задача L. Нагайна

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Нагайна, будучи крестражем Сами-Знаете-Кого, создала армию змей и планирует атаку на Хогвартс.

Вход в школу Хогвартс может быть представлен как прямая линия (ось x) с координатами от 1 до 10^5 . Нагайна запускает змей у входа в Хогвартс. Каждая змея ложится параллельно входу, покрывая некоторый отрезок на расстоянии k от входа от $x = l$ до $x = r$. Формально, каждая змея может быть представлена как отрезок с концами в точках (l, k) и (r, k) . Учтите, что k может быть как положительным, так и отрицательным, но не может быть равно 0.

Пусть для какой-то x -координаты $x = i$, существуют змеи в точке (i, y_1) и в точке (i, y_2) , таких что $y_1 > 0$ и $y_2 < 0$. Тогда если для любой точки (i, y_3) , содержащей змею и такой, что $y_3 > 0$, выполняется $y_1 \leq y_3$, и для любой точки (i, y_4) , содержащей змею и такой, что $y_4 < 0$, выполняется $|y_2| \leq |y_4|$, тогда опасность точки $x = i$ равна $y_1 + |y_2|$. Если не существует таких y_1 или y_2 опасность равна 0.

Гарри хочет посчитать опасность некоторых отрезков входа в Хогвартс. Опасность полуинтервала $[l, r)$ может быть вычислена как сумма величин опасностей во всех целых координатах x этого полуинтервала.

Формально, вам необходимо обрабатывать два типа запросов:

- 1 1 r k: змея появляется у входа между точками $x = l$ и $x = r$ и координатой $y = k$ (включая l , не включая r).
- 2 1 r: вам необходимо узнать опасность полуинтервала с l по r (включая l , не включая r).

Формат входных данных

В первой строке содержится целое число q ($1 \leq q \leq 5 \cdot 10^4$), показывающее число запросов.

Следующие q строк описывают запросы. Описание каждого запроса начинается с типа запроса $type_i$ ($1 \leq type_i \leq 2$). Затем идёт дальнейшее описание запросов. Для запросов типа 1 задаются три целых числа l_i, r_i и k_i ($1 \leq l_i < r_i \leq 10^5$, $-10^9 \leq k_i \leq 10^9$, $k \neq 0$). Иначе задана пара целых чисел l_i и r_i ($1 \leq l_i < r_i \leq 10^5$).

Формат выходных данных

Для каждого запроса типа 2 выведите ответ.

Примеры

стандартный ввод	стандартный вывод
3 1 1 10 10 1 2 4 -7 2 1 10	34
7 1 2 3 5 1 1 10 10 1 4 5 -5 2 4 8 1 1 10 -10 2 4 8 2 1 10	15 75 170

Замечание

ВНИМАНИЕ!!! ПОСЛЕ ЗАПРОСОВ МОЖЕТ НЕ БЫТЬ '\n'