

## Задача А. Треугольники

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 2.5 секунд  
Ограничение по памяти: 256 мегабайт

Дан граф. Найдите количество циклов длины 3.

### Формат входных данных

Первая строка содержит два целых числа  $n$  и  $m$  ( $1 \leq n, m \leq 3 \cdot 10^5$ ) — количество вершин и рёбер, соответственно.

Каждая из следующих  $m$  строк содержит по два целых числа от 1 до  $n$  — вершины, которые соединяет соответствующее ребро.

Гарантируется, что в графе нет петель и кратных рёбер.

### Формат выходных данных

Выведите одно число — ответ.

### Пример

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

## Задача В. Удаление

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 128 мегабайт

Дано  $n$  различных строк  $s_i$ . Требуется найти количество таких пар  $(i, j)$ , что  $i \neq j$ , а так же  $s_j$  можно получить из  $s_i$  каждый раз удаляя только первый или второй символ из  $s_i$ .

Например из строки `abcxux` можно получить строку `сux`: `abcxux`  $\rightarrow$  `асхux`  $\rightarrow$  `схux`  $\rightarrow$  `сux`.

### Формат входных данных

Первая строка содержит число  $n$  ( $2 \leq n \leq 200000$ ) — количество строк.

$i$ -я из следующих  $n$  строк содержит строку  $s_i$  ( $1 \leq |S|$ ). Гарантируется, что  $s_i$  состоит из строчных букв латинского алфавита.

Так же гарантируется, что  $|s_1| + \dots + |s_n| \leq 10^6$ .

### Примеры

стандартный ввод	стандартный вывод
3 abcxux сux abc	1
6 b a abc c d ab	5

## Задача С. Профили-двойники

Имя входного файла: `stdin`  
Имя выходного файла: `stdout`  
Ограничение по времени: 1 second  
Ограничение по памяти: 256 megabytes

Вам предложили работу в компании, разрабатывающей крупную социальную сеть. Ваше первое задание связано с поиском профилей, с большой вероятностью принадлежащих одному и тому же пользователю.

В социальной сети зарегистрировано  $n$  профилей, пронумерованных от 1 до  $n$ . Некоторые пары среди них являются друзьями (отношение «быть друзьями» взаимно, то есть если  $i$  является другом  $j$ , то и  $j$  является другом  $i$ ). Будем говорить, что профили  $i$  и  $j$  ( $i \neq j$ ) являются *двойниками*, если для любого профиля  $k$  ( $k \neq i, k \neq j$ ), верно одно из двух утверждений: либо  $k$  дружит и с  $i$ , и с  $j$ , либо  $k$  не дружит ни с одним из них. При этом  $i$  и  $j$  могут как дружить между собой, так и не дружить.

Вам нужно посчитать количество различных неупорядоченных пар  $(i, j)$ , таких что профили  $i$  и  $j$  — двойники. Обратите внимание, что пары неупорядоченные, то есть пары  $(a, b)$  и  $(b, a)$  считается одинаковыми.

### Формат входных данных

В первой строке записано два целых числа  $n$  и  $m$  ( $1 \leq n \leq 10^6, 0 \leq m \leq 10^6$ ), разделенных пробелом — количество профилей и количество пар друзей соответственно.

В следующих  $m$  строках записаны описания пар друзей в формате « $v$   $u$ », где  $v$  и  $u$  ( $1 \leq v, u \leq n, v \neq u$ ) — номера профилей, являющихся друзьями. Гарантируется, что каждая неупорядоченная пара друзей встречается не более одного раза и никакой профиль не является другом самого себя.

### Формат выходных данных

Выведите единственное целое число — количество неупорядоченных пар профилей, являющихся двойниками.

### Примеры

<code>stdin</code>	<code>stdout</code>
3 3 1 2 2 3 1 3	3
3 0	3
4 1 1 3	2

## Задача D. Хеш-код

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

Согласно документации стандартной библиотеки Java, хеш-код для строки вычисляется как:

$$s[0] \cdot 31^{n-1} + s[1] \cdot 31^{n-2} + \dots + s[n-1]$$

Где  $s[i]$  — это  $i$ -й символ строки,  $n$  длина строки. Для вычисления используются целые 32-битные числа в форме дополнения до двух.

Вы собираетесь взломать сервера одной известной компании. Чтобы вы смогли выполнить атаку, вам нужны  $k$  различных строк, которые имеют одинаковые хеш-коды. К сожалению, сервера этой компании не принимают строки запроса, содержащие буквы отличные от английских букв нижнего и верхнего регистров.

Напишите программу, которая генерирует такие строки.

### Формат входных данных

Первая строка содержит целое число  $k$  — количество необходимых строк запроса для генерации ( $2 \leq k \leq 1000$ ).

### Формат выходных данных

Необходимо вывести  $k$  различных непустых строк, каждая из которых имеет длину не более 1000 символов. Все строки должны состоять только из английских букв верхнего или нижнего регистров и иметь равный хеш-код.

### Пример

стандартный ввод	стандартный вывод
4	edHs mENAGeS fEHs edIT

## Задача Е. Изоморфизм деревьев

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

Даны два дерева, состоящие из  $n$  вершин. Требуется проверить, изоморфны ли эти деревья друг другу.

### Формат входных данных

Первая строка содержит целое число  $n$  ( $1 \leq n \leq 10^5$ ) — количество вершин в каждом из деревьев.

Каждая из следующих  $n - 1$  строк содержит два целых числа  $a_i$  и  $b_i$  ( $1 \leq a_i, b_i \leq n$ ) — описание  $i$ -го ребра первого дерева. Гарантируется, что данная последовательность ребер задает дерево.

Каждая из следующих  $n - 1$  строк содержит два целых числа  $c_i$  и  $d_i$  ( $1 \leq c_i, d_i \leq n$ ) — описание  $i$ -го ребра второго дерева. Гарантируется, что данная последовательность ребер задает дерево.

### Формат выходных данных

Выведите слово «YES», если деревья изоморфны друг другу. В противном случае выведите слово «NO».

### Примеры

стандартный ввод	стандартный вывод
6 1 3 2 3 3 4 4 5 4 6 1 2 1 5 1 6 2 4 2 3	YES
4 1 2 1 3 1 4 1 2 2 3 3 4	NO

## Задача F. Компоненты связности на отрезке

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

Наде на выпускной вместе с аттестатом подарили список ребер графа на  $n$  вершинах.

Отметив окончание школы и вернувшись домой, она решила заняться своим любимым делом: поотвечать на запросы на отрезке. Тут она вспомнила про подарок, и решила, что будет отвечать на следующие запросы: «если рассмотреть граф на  $n$  вершинах и ребрах, находящихся с  $l$ -го по  $r$ -е в списке ребер, то сколько компонент связности будет в этом графе?»

Впрочем, было уже почти утро, и ей быстро надоел этот процесс, поэтому она пошла спать, а ответы на запросы перепоручила вам...

### Формат входных данных

На первой строке находятся два целых числа  $n$  и  $m$  ( $1 \leq n, m \leq 10^5$ ) — количество вершин и ребер в графе соответственно.

На следующих  $m$  строках находятся по два целых числа  $u_i$  и  $v_i$  ( $1 \leq u_i, v_i \leq n$ ) — вершины, которые соединяет  $i$ -е ребро.

На следующей строке находится целое число  $q$  ( $1 \leq q \leq 10^5$ ) — количество запросов.

На следующих  $q$  строках находятся по два целых числа  $l_i$  и  $r_i$  ( $1 \leq l_i, r_i \leq n$ ) — границы  $i$ -го запроса.

### Формат выходных данных

Для каждого запроса на отдельной строке выведите ответ на него.

### Пример

стандартный ввод	стандартный вывод
4 4	3
2 3	2
1 2	1
3 4	1
4 1	
4	
2 2	
2 3	
1 4	
2 4	

## Задача G. Частота строки

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 1.5 секунд  
Ограничение по памяти: 512 мегабайт

Дана строка  $s$ . Требуется ответить на  $n$  запросов.  $i$ -й запрос состоит из целого числа  $k_i$  и строки  $m_i$ , ответом является минимальная длина строки  $t$  такой, что  $t$  является подстрокой  $s$  и строка  $m_i$  входит в  $t$  как подстрока не менее  $k_i$  раз.

Подстрокой строки называется любая последовательность подряд идущих символов в этой строке.

Гарантируется, что для любых двух запросов строки  $m_i$  из этих запросов различны.

### Формат входных данных

В первой строке содержится строка  $s$  ( $1 \leq |s| \leq 10^5$ ).

Во второй строке содержится целое число  $n$  ( $1 \leq n \leq 10^5$ ).

В каждой из следующих  $n$  строк содержатся целое число  $k_i$  ( $1 \leq k_i \leq |s|$ ) и непустая строка  $m_i$  — параметры запроса с номером  $i$ .

Все строки во вводе состоят только из строчных букв латинского алфавита. Суммарная длина всех строк во вводе не превосходит  $10^5$ . Все  $m_i$  различны.

### Формат выходных данных

Для каждого запроса выведите ответ на него в отдельной строке.

Если строка  $m_i$  встречается в  $s$  менее  $k_i$  раз, выведите  $-1$ .

### Примеры

стандартный ввод	стандартный вывод
aaaaa	3
5	4
3 a	4
3 aa	-1
2 aaa	5
3 aaaa	
1 aaaaa	
abbb	-1
7	2
4 b	-1
1 ab	3
3 bb	-1
1 abb	1
2 bbb	-1
1 a	
2 abbb	

## Задача Н. Польшар и Подарки

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

Рождество! Польшар и его друзья будут дарить друг другу подарки. Всего шаров  $n$ . Каждый шар должен подарить подарок ровно одному другому шару в соответствии с некоторой перестановкой  $p$ ,  $p_i \neq i$  для всех  $i$ .

К сожалению, шары забывчивы. Мы знаем, что ровно  $k$  шаров забудут принести свои подарки. Шар номер  $i$  получит подарок, если будут выполнены следующие два условия:

1. Шар номер  $i$  должен принести свой подарок.
2. Шар  $x$  такой, что  $p_x = i$ , должен принести свой подарок.

Какое минимально и максимально возможное число шаров, которые **не** получают свой подарок, если ровно  $k$  шаров забудут принести свой подарок?

### Формат входных данных

В первой строке находится два целых числа  $n$  и  $k$  ( $2 \leq n \leq 10^6$ ,  $0 \leq k \leq n$ ) — общее число шаров и число шаров, которые забудут подарки.

Во второй строке находится перестановка  $p$  целых чисел от 1 до  $n$ , где  $p_i$  — номер шара, которому должен дать подарок шар номер  $i$ . Для всех  $i$  выполняется  $p_i \neq i$ .

### Формат выходных данных

Выведите два числа — минимально и максимально возможное число шаров, которые **не** получают подарков, соответственно.

### Примеры

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

### Замечание

В первом примере, если первый и третий шары забудут принести подарок, то они же и будут единственными, кто не получит подарка. Поэтому минимальный ответ равен 2. Однако, если первый и второй шары забудут, то только пятый шар получит подарок. Поэтому максимальный ответ равен 4.



## Задача I. Запросы Таро на дереве

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	6 секунд
Ограничение по памяти:	256 мегабайт

У кота Таро есть подвешенное дерево из  $n$  вершин. Вершины пронумерованы от  $0$  до  $n - 1$ , включительно. Вершина с номером  $0$  является корнем дерева. На каждом ребре написан целый положительный вес.

Таро хочет выполнить  $m$  запросов. Каждый запрос состоит из двух вершин  $u$  и  $v$ . Запросы следует выполнить следующим образом:

- Если  $u$  равно  $v$ , выведите  $-1$ .
- Иначе, если вершина  $v$  лежит в поддереве вершины  $u$ , выведите максимальный вес на ребрах на простом пути из  $v$  в  $u$ .
- Иначе, удалите ребро, соединяющее вершину  $u$  и ее родителя. Вместо него добавьте новое ребро (с таким же весом), которое сделает вершину  $u$  ребенком вершины  $v$ . (Обратите внимание, что все поддерево вершины  $u$  теперь часть поддерева вершины  $v$ .) Ничего выводить не нужно.

### Формат входных данных

В первой строке содержатся два целых числа  $n$  и  $m$  ( $1 \leq n, m \leq 200\,000$ ) — количество вершин и запросов, соответственно.

Во второй строке содержится  $n - 1$  целое число  $p_i$  ( $1 \leq i \leq n - 1, 0 \leq p_i < i$ ) — родитель  $i$ -й вершины.

В третьей строке содержится  $n - 1$  целое число  $w_i$  ( $1 \leq i \leq n - 1, 0 \leq w_i \leq 1\,000\,003$ ) — вес ребра из вершины  $i$  в ее родителя.

В следующих  $m$  строках содержится по два целых числа  $u_i$  и  $v_i$  ( $0 \leq v_i, u_i \leq n - 1$ ) — запросы.

### Формат выходных данных

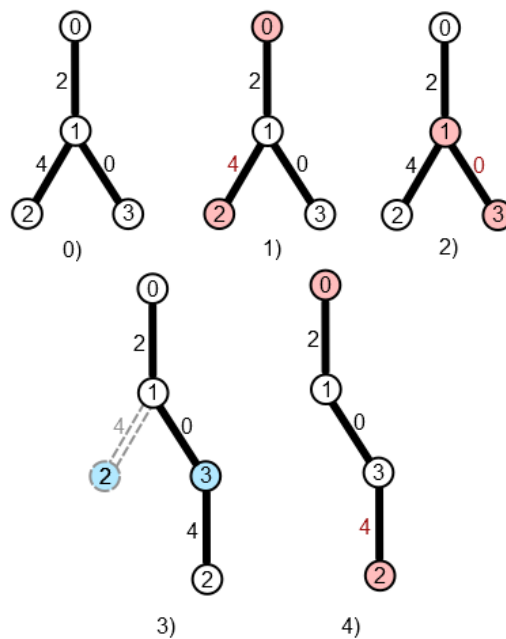
Выведите ответы на запросы.

## Примеры

стандартный ввод	стандартный вывод
4 4 0 1 1 2 4 0 0 2 1 3 2 3 0 2	4 0 4
7 10 0 1 1 2 4 4 5 4 0 2 4 6 0 1 1 1 2 6 2 4 2 5 0 2 0 2 1 5 1 5 2 6	5 -1 6 2 4 5 5 4 4 4 6
10 4 0 0 1 2 3 5 6 7 8 14 24 25 65 13 59 19 37 58 1 9 5 6 2 4 6 8	59 59 65 37

## Замечание

Пояснение к первому примеру.



## Задача J. Грибные пары

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мегабайт

Недавно столичный Центр Помощи Мигрантам открыл свою школу. Школа эта очень странная — занятия идут не по 45 минут, ученики пропускают уроки, а ведет их учитель по прозвищу ГРИБ. В один момент это надоело всем (кроме учеников, конечно же), из-за чего учитель решил проучить всех прогульщиков.

ГРИБ дал ученикам следующую задачу: Дан массив  $a$ , состоящий из  $n$  целых чисел. К этому массиву приходят  $m$  запросов, состоящих из двух чисел  $x_i$  и  $y_i$ . Для каждого запроса требуется максимизировать произведение числа вхождений  $x_i$  до некоторой позиции на количество вхождений  $y_i$  начиная с этой позиции. Более формально, можно ввести следующие три функции:

- $lcnt(j, x)$  — количество вхождений числа  $x$  на префиксе массива  $a$  до позиции  $j$  включительно.
- $rcnt(j, x)$  — количество вхождений числа  $x$  на суффиксе массива  $a$  начиная с позиции  $j$  включительно.
- $f(i, x, y) = lcnt(i - 1, x) \cdot rcnt(i, y)$

Для каждого запроса необходимо по всем  $j$  от 2 до  $n$  найти максимум  $f(j, x_i, y_i)$ . Так как ученики пропустили все занятия, они не могут решить задачу ГРИБ'а. Помогите ученикам школы Центра Помощи Мигрантам решить эту задачу и избежать отчисления из школы.

### Формат входных данных

В первой строке даны два целых числа  $n$  и  $m$  ( $2 \leq n \leq 100\,000, 1 \leq m \leq 100\,000$ ) — количество чисел в массиве и число запросов.

Во второй строке даны  $n$  целых чисел  $a_1, a_2, \dots, a_n$  ( $1 \leq a_i \leq 10^9$ ) — числа в массиве.

В следующих  $m$  строках описаны запросы. В каждой из них даны два целых числа  $x_i$  и  $y_i$  ( $1 \leq x_i, y_i \leq 10^9$ ) — значения из  $i$ -го запроса. Гарантируется, что числа  $x_i$  и  $y_i$  присутствуют в массиве.

### Формат выходных данных

В  $m$  строках выведите ответы на запросы, по одному в строке.

### Примеры

стандартный ввод	стандартный вывод
5 3 1 2 3 2 1 1 2 2 2 1 2	2 1 2
5 4 1 1 1 2 2 1 1 1 2 2 2 2 1	2 6 1 0

### Замечание

Рассмотрим первый пример.  
Первый запрос — 1 2:

- $f(2, 1, 2) = 2$
- $f(3, 1, 2) = 1$
- $f(4, 1, 2) = 1$
- $f(5, 1, 2) = 0$

Таким образом, ответ на первый запрос равен 2.  
Второй запрос — 2 2:

- $f(2, 2, 2) = 0$
- $f(3, 2, 2) = 1$
- $f(4, 2, 2) = 1$
- $f(5, 2, 2) = 0$

Таким образом, ответ на второй запрос равен 1.  
Третий запрос совпадает с первым, и ответ на него равен 2.

## Задача К. Подарок

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

У Пети есть набор из  $n$  карточек, на которых написаны положительные целые числа. Он хочет подарить какие-то из этих карточек Васе. Чтобы подарок выглядел солидно, Петя хочет, чтобы произведение чисел на подаренных карточках было не меньше  $k$ . Сколько различных способов выбрать подарок есть у Пети? (способы считаются различными, если есть карточка, которая есть в одном наборе, но нет в другом). Так как ответ может быть слишком большим, выведите его по модулю  $10^9 + 7$ .

### Формат входных данных

Первая строка содержит числа  $n$  и  $k$ . ( $1 \leq n \leq 1000$ ,  $2 \leq k \leq 10^9$ ). Вторая строка содержит числа  $a_i$ , написанные на карточках ( $1 \leq a_i \leq 10^9$ ).

### Формат выходных данных

Выведите число способов выбрать подарок по модулю  $10^9 + 7$ .

### Примеры

стандартный ввод	стандартный вывод
3 5 2 3 10	5
5 10 2 2 7 1 3	16

## Задача L. Чтение строк

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

Дано дерево  $T$  состоящее из  $n$  вершин (пронумерованных целыми числами от 1 до  $n$ ). В каждой вершине записана некоторая буква. Корень дерева расположен в вершине 1.

Рассмотрим поддерево  $T_v$  некоторой вершины  $v$ . Вдоль любого простого пути, начинающегося в  $v$  и заканчивающегося в некоторой вершине  $u \in T_v$  (возможно, в самой  $v$ ), можно прочитать некоторую строку. Обозначим количество **различных** строк, которые можно прочитать таким способом как  $\text{dif}(v)$ .

Дополнительно: для каждой вершины  $v$  дано целое число  $c_v$ . Нас интересуют вершины, в которых значение  $\text{dif}(v) + c_v$  как можно больше.

Вы должны вычислить две величины — максимальное значение  $\text{dif}(v) + c_v$  и количество вершин  $v$  с максимальным  $\text{dif}(v) + c_v$ .

### Формат входных данных

В первой строке входных данных записано единственное целое число  $n$  ( $1 \leq n \leq 300\,000$ ) — количество вершин в дереве  $T$ .

Во второй строке записано  $n$  целых чисел  $c_i$  ( $0 \leq c_i \leq 10^9$ ).

В третьей строке записана строка  $s$ , состоящая из  $n$  строчных букв английского алфавита, —  $i$ -й символ этой строки соответствует букве, записанной в вершине  $i$ .

Далее следует  $n - 1$  строка с описанием рёбер дерева  $T$ . Каждая из них содержит два целых числа  $u$  и  $v$  ( $1 \leq u, v \leq n$ ), обозначающих ребро, которое соединяет вершины  $u$  и  $v$ .

Гарантируется, что входные данные описывают дерево.

### Формат выходных данных

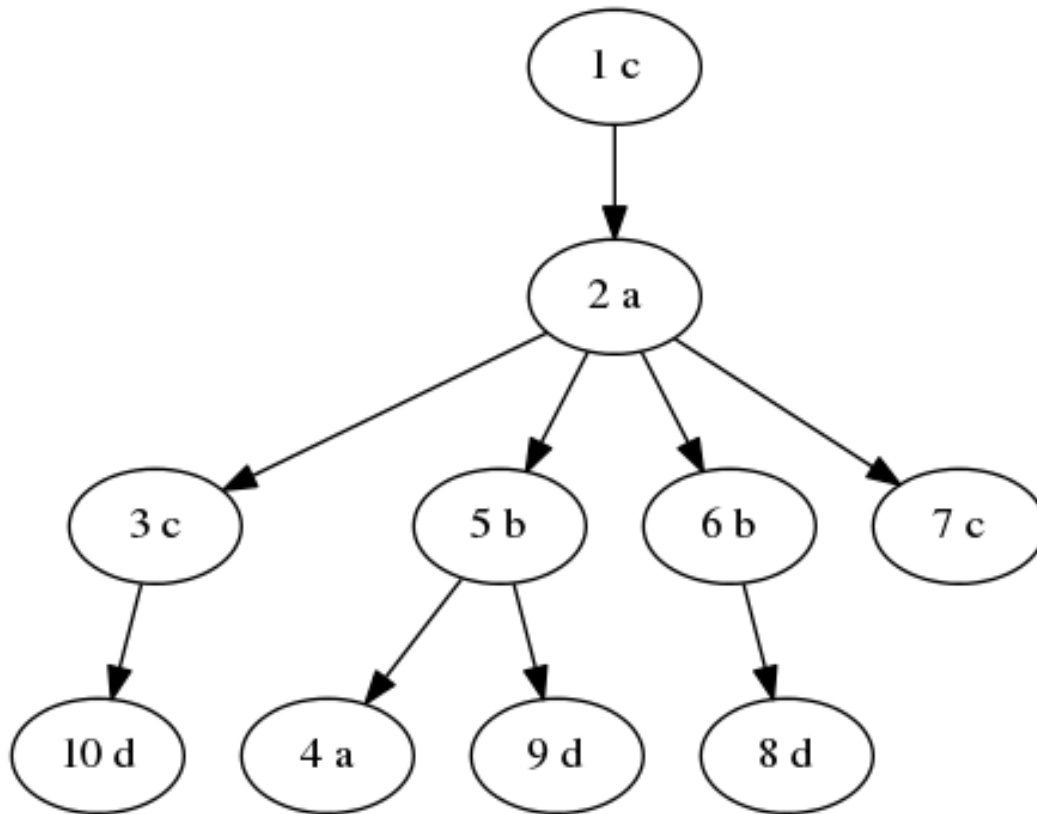
Выведите два числа — значение  $m = \max(\text{dif}(i) + c_i)$  для всех  $1 \leq i \leq n$  и количество вершин  $v$ , для которых  $m = \text{dif}(v) + c_v$ .

### Примеры

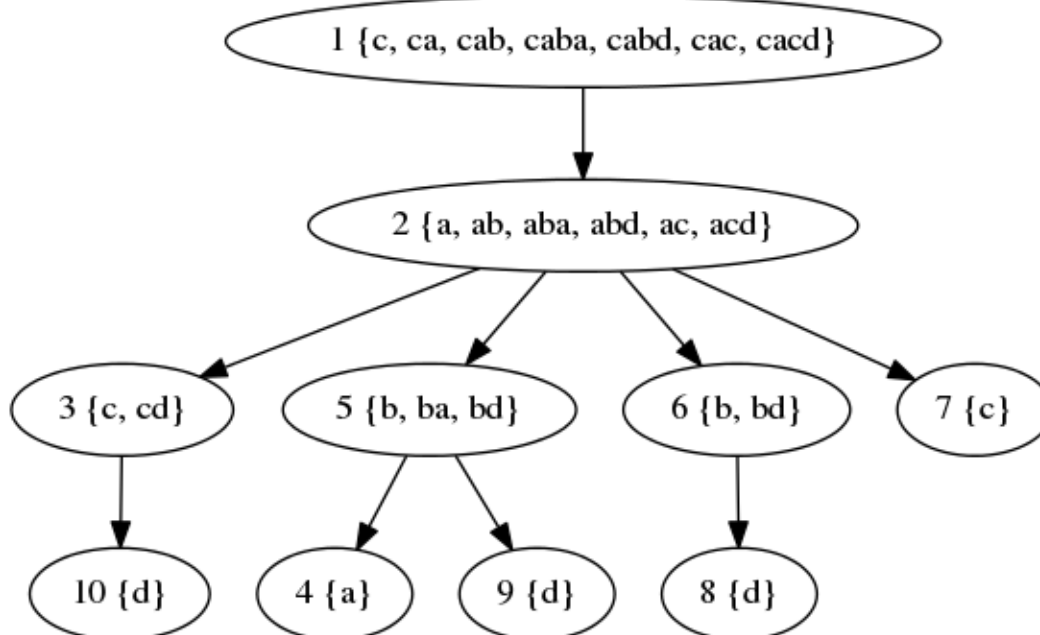
стандартный ввод	стандартный вывод
10 1 2 7 20 20 30 40 50 50 50 cacabbccddd 1 2 6 8 7 2 6 2 5 4 5 9 3 10 2 5 2 3	51 3
6 0 2 4 1 1 1 raaaba 1 2 2 3 2 4 2 5 3 6	6 2

## Замечание

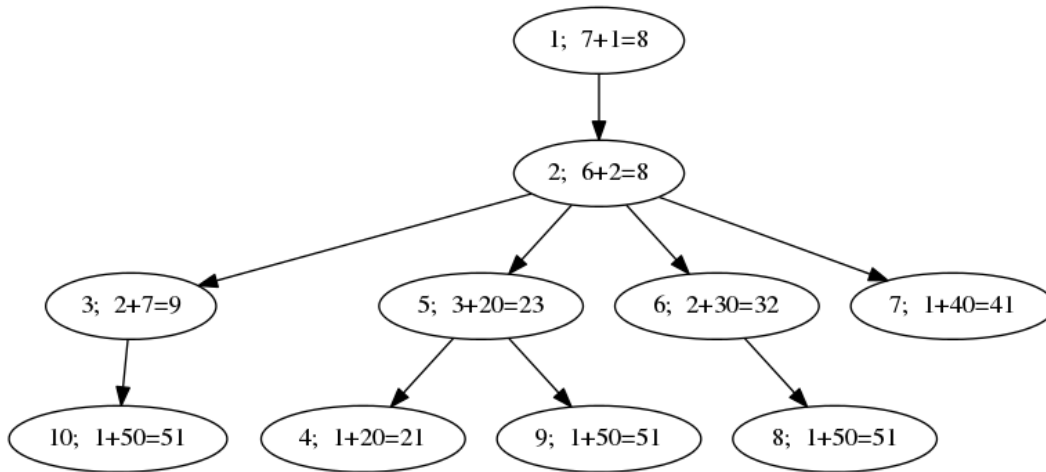
В первом примере дерево выглядит следующим образом:



Наборы строк, которые могут быть прочитаны из вершин:



Наконец, значения  $\text{dif}(v) + c_v$  таковы:



Во втором примере значения  $\text{dif}(1..n)$  таковы:  $(5, 4, 2, 1, 1, 1)$ . Различные строки, которые можно прочитать из вершины 2 таковы:  $a, aa, aaa, ab$ ; обратите внимание, что  $aa$  может быть прочитано как на пути до вершины 3, так и на пути до вершины 4.



## Задача М. Массивы-палиндромы

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

Кай работает в лаборатории изучения массивов, он экспериментирует с двумя массивами натуральных чисел:  $A = [a_1, a_2, \dots, a_n]$  длины  $n$  и  $B = [b_1, b_2, \dots, b_m]$  длины  $m$ .

Эксперимент, который проводит Кай, устроен следующим образом. У каждого из массивов отбрасывается произвольный, возможно пустой, префикс, а также произвольный, возможно пустой, суффикс, таким образом, чтобы оставшиеся части массивов имели равную длину. Обозначим получившиеся массивы как  $A'$  и  $B'$ , а их длину как  $k$ . Затем Кай суммирует поэлементно получившиеся массивы, итоговый массив Кай обозначает как  $C = [c_1, c_2, \dots, c_k]$ .

Пусть, например,  $n = 5$ ,  $A = [4, 3, 3, 2, 1]$ ,  $m = 6$ ,  $B = [4, 1, 5, 1, 3, 2]$ , от массива  $A$  отбрасывается первый и последний элемент, от массива  $B$  три первых. После этого массивы имеют вид  $A' = [3, 3, 2]$ ,  $B' = [1, 3, 2]$ , результат их поэлементного суммирования  $C = [4, 6, 4]$ .

Задача Кая заключается в том, чтобы получать такие  $C$ , которые являются *массивами-палиндромами*, то есть если числа на первой и последней позиции совпадают, числа на второй и предпоследней позиции совпадают, и так далее, для всех  $i$  числа на позициях  $i$  и  $k - i + 1$  совпадают.

Помогите Каю понять, какой максимальный по длине массив-палиндром он может получить в результате эксперимента.

### Формат входных данных

В первой строке ввода даны два целых числа  $n$  и  $m$  — количество элементов в первом и во втором массиве, соответственно ( $1 \leq n, m \leq 100\,000$ ).

Во второй строке ввода даны  $n$  целых чисел  $a_i$  — массив  $A$  ( $1 \leq a_i \leq 100$ ).

В третьей строке ввода даны  $m$  целых чисел  $b_j$  — массив  $B$  ( $1 \leq b_j \leq 100$ ).

### Формат выходных данных

Выведите единственное целое число — максимальное  $k$ , что Кай в результате эксперимента может получить массив-палиндром длины  $k$ .

### Пример

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

## Задача N. Путешествие

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	32 мегабайта

Максат живет в стране где есть  $N$  городов. Города соединены  $N - 1$  дорогами одинаковой длины. Из каждого города можно добраться до любого другого города. В каждом городе живет некоторое количество жителей. Максат каждую неделю выбирает два города и идет по кратчайшему пути от одного города до другого. В своем пути он выбирает два города, таких что общее количество жителей в этих двух городах равно  $C$ . Вы хотите посчитать сколькими способами он мог выбрать два города на этом пути таких, что общее количество жителей в этих двух городах равно  $C$ .

### Формат входных данных

В первой строке дано количество городов  $N$ , во второй строке количество жителей в каждом городе  $a_i (1 \leq a_i \leq 10^9)$ .

В следующих  $N - 1$  строках даны пары чисел, обозначающих города соединенные дорогой.

В следующей строке даны два числа — количество путешествий  $M$  и общее количество жителей в двух городах  $C (1 \leq C \leq 10^9)$ .

В следующих  $M$  строках даны пары городов в которых Максат начал и закончил свой путь.

### Формат выходных данных

Для каждого путешествия выведите ответ.

### Система оценки

Данная задача содержит 4 подзадачи:

- $1 \leq N \leq 3, M = 1$ . Оценивается в 10 баллов.
- $1 \leq N \leq 300, 1 \leq M \leq 300$ . Оценивается в 15 баллов.
- $1 \leq N \leq 5000, 1 \leq M \leq 5000$ . Оценивается в 25 баллов.
- $1 \leq N \leq 50000, 1 \leq M \leq 50000$ . Оценивается в 50 баллов.

### Примеры

стандартный ввод	стандартный вывод
3 10 20 30 1 2 2 3 1 40 1 3	1
3 10 10 10 1 2 2 3 1 20 1 3	3

## Задача О. Глеб и медиана

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1.3 секунд
Ограничение по памяти:	512 мегабайт

Глеб устал от побитового исключающего «или» и решил, что пора найти новую интересную функцию. Его выбор пал на медиану. Напомним, медианой массива называется число, которое окажется посередине, если массив упорядочить по возрастанию. В рамках этой задачи для массивов чётной длины положим медиану равной левому из двух центральных в отсортированном порядке элементов.

Для некоторого числа  $m$  назовём  $m$ -разбиением массива такое его разбиение на непересекающиеся отрезки, что на каждом из этих отрезков медиана больше либо равна  $m$ . Вам дан массив  $a$  длины  $n$  и  $q$  запросов двух видов:

1. присвоить элементу с индексом  $i$  значение  $x$ ;
2. найти наибольшее число  $k$  такое, что для подотрезка массива с индексами от  $l$  до  $r$  существует  $m$ -разбиение на  $k$  отрезков.

### Формат входных данных

В первой строке дается число  $n$  ( $1 \leq n \leq 2 \cdot 10^5$ ) - размер массива. В следующей строке вводятся  $n$  чисел  $a_i$  ( $1 \leq a_i \leq 10^9$ ) - элементы массива, на следующей строке вводится число  $q$  ( $1 \leq q \leq 2 \cdot 10^5$ ) - количество запросов. В следующих  $q$  строках даются запросы, каждый в одном из следующих форматов:

- $1 \ i \ x$  — запрос 1 типа ( $1 \leq i \leq n, 1 \leq x \leq 10^9$ );
- $2 \ m \ l \ r$  — запрос 2 типа ( $1 \leq m \leq 10^9, 1 \leq l \leq r \leq n$ ).

### Формат выходных данных

Для каждого запроса второго типа в отдельной строке выведите ответ на запрос. В случае если для отрезка не существует никакого  $m$ -разбиения, выведите 0.

### Пример

стандартный ввод	стандартный вывод
5	1
1 2 3 4 5	0
4	2
2 2 1 3	
1 1 5	
2 5 1 5	
2 4 4 5	

### Замечание

Даже не пытайтесь тут записать что-то кроме корневой

## Задача Р. Машинное обучение

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

На курсе машинного обучения вам выдали первое домашнее задание — вам предстоит проанализировать некоторый массив из  $n$  чисел.

В частности, вы интересуетесь так называемой *равномерностью* массива. Предположим, что в массиве число  $b_1$  встречается  $k_1$  раз,  $b_2$  —  $k_2$  раз, и т.д. Тогда *равномерностью* массива называется такое минимальное целое число  $c \geq 1$ , что  $c \neq k_i$  для любого  $i$ .

В рамках вашего исследования вы хотите последовательно проделать  $q$  операций.

- Операция  $t_i = 1, l_i, r_i$  задаёт запрос исследования. Необходимо вывести равномерность массива, состоящего из элементов на позициях от  $l_i$  до  $r_i$  включительно.
- Операция  $t_i = 2, p_i, x_i$  задаёт запрос уточнения данных. Начиная с этого момента времени  $p_i$ -му элементу массива присваивается значения  $x_i$ .

### Формат входных данных

Первая строка содержит  $n$  и  $q$  ( $1 \leq n, q \leq 100\,000$ ) — размер массива и число запросов соответственно.

Во второй строке записаны ровно  $n$  чисел —  $a_1, a_2, \dots, a_n$  ( $1 \leq a_i \leq 10^9$ ).

Каждая из оставшихся  $q$  строк задаёт очередной запрос.

Запрос первого типа задаётся тремя числами  $t_i = 1, l_i, r_i$ , где  $1 \leq l_i \leq r_i \leq n$  — границы соответствующего отрезка.

Запрос второго типа задаётся тремя числами  $t_i = 2, p_i, x_i$ , где  $1 \leq p_i \leq n$  — позиция в которой нужно заменить число, а  $1 \leq x_i \leq 10^9$  — его новое значение

### Формат выходных данных

Для каждого запроса первого типа выведите одно число — равномерность соответствующего отрезка массива.

### Пример

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

### Замечание

Первый запрос состоит из ровно одного элемента — 1. Минимальное подходящее  $c = 2$ .

Отрезок второго запроса состоит из четырёх 2, одной 3 и двух 1. Минимальное подходящее  $c = 3$ .

Отрезок четвёртого запроса состоит из трёх 1, трёх 2 и одной 3. Минимальное подходящее  $c = 2$ .

## Задача Q. Простые xor-разбиения

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 10 секунд  
Ограничение по памяти: 512 мегабайт

Вам дан массив натуральных чисел. Ваша задача заключается в том, чтобы обрабатывать два типа запросов:

- Изменить значение элемента массива
- По заданному небольшому множеству чисел сказать, можно ли разбить весь массив на какое-то количество непересекающихся подотрезков, что на каждом отрезке «исключающее или» (XOR) значений принадлежит этому множеству. Каждый элемент массива должен принадлежать какому-то отрезку.

Заметьте, что не обязательно, чтобы каждое число из заданного множества было равно «исключающему или» какого-либо из отрезков, а также любое число из множества может быть равно «исключающему или» любого количества подотрезков. Например, если «исключающее или» всех чисел в массиве принадлежит множеству, можно выбрать весь массив в качестве единственного подотрезка.

### Формат входных данных

Первая строка содержит два числа  $n$  и  $q$  ( $1 \leq n \leq 10^5$ ;  $1 \leq q$ ) — длина массива и количество запросов.

Вторая строка содержит  $n$  чисел  $x_i$  ( $0 \leq x_i < 2^{20}$ ) — изначальный массив. Следующие  $q$  строк описывают запросы. Каждый запрос может быть одного из двух типов.

Запросы первого типа задаются в формате  $1 \ a \ b$  ( $1 \leq a \leq n$ ;  $0 \leq b < 2^{20}$ )? где  $a$  это индекс изменяемого элемента, а  $b$  это его новое значение.

Запросы второго типа задаются в формате  $2 \ k \ a_1 \ \dots \ a_k$  ( $1 \leq k \leq 5$ ;  $0 \leq a_i < 2^{20}$ ) где  $k$  это размер множества, а  $a_i$  это его элементы. Заметьте, что числа  $a_i$  не обязательно различны.

Количество запросов первого типа не превосходит  $4 \cdot 10^5$ , и сумма значений  $k$  по всем запросам второго типа не превосходит  $10^5$ .

### Формат выходных данных

Для каждого запроса второго типа выведите одну строку в таком же порядке, как они следуют во входе. Если разбить массив в соответствующем запросе возможно, выведите **ТАК** («да» по-польски), иначе выведите **НIE** («нет» по-польски).

### Пример

стандартный ввод	стандартный вывод
5 10	ТАК
1 2 0 3 0	ТАК
2 1 3	ТАК
2 1 0	НIE
1 3 5	
2 2 6 3	
1 1 8	
1 2 5	
1 3 3	
1 4 1	
1 5 1	
2 3 2 4 8	

## Замечание

В первом запросе второго типа массив равен  $\{1, 2, 0, 3, 0\}$ , поэтому его можно разбить следующим образом:  $\{1, 2, 0\}$  и  $\{3, 0\}$ .

Во втором запросе первого типа массив такой же, и «исключающее или» всех элементов равен нулю, поэтому можно выбрать весь массив в качестве подотрезка.

В третьем запросе второго типа массив равен  $\{1, 2, 5, 3, 0\}$ , и его можно разбить на  $\{1, 2, 5\}$  и  $\{3, 0\}$ .

В последнем запросе разбить массив невозможно.

## Задача R. Сериал по Майнкрафту

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	6 секунд
Ограничение по памяти:	512 мегабайт

Маленький Миша ходит на кружок по программированию и ничего там не решает. Это может показаться странным, но когда вы узнаете, что Миша снимает сериал по Майнкрафту, все сразу встанет на свои места...

Миша, вдохновляясь застройкой Манхэттена, построил в Майнкрафте город, который можно представить в виде таблицы  $n \times m$ . В городе живут  $k$  школьников,  $i$ -й школьник живет в доме, который находится на пересечении  $x_i$ -й строки и  $y_i$ -го столбца. Также у каждого школьника есть степень его агрессивности  $w_i$ . Так как город оказался очень большим, Миша решил территориально ограничить действия своего сериала некоторым принадлежащим таблице квадратом  $s$ , стороны которого параллельны осям координат и имеют длину от 1 до  $\min(n, m)$  клеток.

По сюжету главный герой приедет в город и сразу же попадет в квадрат  $s$ . Обладая уникальной степенью агрессивности 0 он сможет проявить свои лидерские качества и собрать команду из спокойных, умеренных и агрессивных школьников.

Чтобы собранная команда была разносторонней, но сплоченной, агрессивности всех школьников в ней должны быть попарно различны и должны образовывать единый отрезок подряд идущих целых чисел. То есть, если внутри квадрата  $s$  **найдутся** школьники со степенями агрессивности  $l, l + 1, \dots, -1, 1, \dots, r - 1, r$ , где  $l \leq 0 \leq r$ , то главный герой сможет собрать команду из  $r - l + 1$  человека (сам он тоже входит в эту команду).

**Обратите внимание**, брать в команду всех школьников из квадрата  $s$  не обязательно.

Миша считает, что в команде главного героя должно быть хотя бы  $t$  человек. Поэтому его интересует, сколько существует квадратов в таблице, попав в которые, главный герой сможет набрать команду как минимум из  $t$  человек. Помогите Мише это посчитать.

### Формат входных данных

Первая строка содержит четыре целых числа  $n, m, k$  и  $t$  ( $1 \leq n, m \leq 40\,000$ ,  $1 \leq n \cdot m \leq 40\,000$ ,  $1 \leq k \leq 10^6$ ,  $1 \leq t \leq k + 1$ ) — количество строк в таблице, количество столбцов, количество школьников, которые живут в городе и необходимый размер команды соответственно.

Каждая из  $k$  следующих строк содержит по три целых числа  $x_i, y_i$  и  $w_i$  ( $1 \leq x_i \leq n$ ,  $1 \leq y_i \leq m$ ,  $1 \leq |w_i| \leq 10^9$ ) — номер строки и номер столбца, на пересечении которых живет  $i$ -й школьник, а так же степень его агрессивности.

### Формат выходных данных

Выведите одно целое число — количество способов выбрать квадрат  $s$  таким образом, чтобы главный герой смог набрать команду, состоящую, хотя бы из  $t$  человек.

### Примеры

стандартный ввод	стандартный вывод
2 2 1 2 1 1 2	0
2 2 2 2 1 1 1 2 2 2	2
2 2 4 2 1 1 1 1 1 -1 1 2 1 2 2 1	4

### Замечание

1. В первом тестовом примере главный герой ни в каком выбранном квадрате  $s$  не сможет набрать команду, состоящую из более чем одного человека.

2	

Иллюстрация к первому тестовому примеру.

2. Во втором тестовом примере можно выбрать квадрат  $s$  двумя способами, изображенными ниже, в одном из них главный герой сможет набрать команду из школьников с степенями агрессивности  $[0, 1]$ , а в другом —  $[0, 1, 2]$ . Обратите внимание на то, что вне зависимости от выбранного квадрата главный герой со степенью агрессивности 0 всегда будет входить в команду.

<b>1</b>	
	2

<b>1</b>	
	<b>2</b>

Иллюстрация ко второму тестовому примеру.

3. В третьем тестовом примере можно выбрать квадрат  $s$  четырьмя способами, изображенными ниже, в них главный герой сможет набрать команды со следующими степенями агрессивности школьников, соответственно:  $[-1, 0, 1]$ ,  $[0, 1]$ ,  $[0, 1]$ ,  $[-1, 0, 1]$ .

<b>1,-1</b>	<b>1</b>
	1

1,-1	<b>1</b>
	1

1,-1	1
	<b>1</b>

<b>1,-1</b>	<b>1</b>
	1

Иллюстрация к третьему тестовому примеру.



## Задача S. Жесть

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 15 секунд  
Ограничение по памяти: 256 мегабайт

Дан массив из  $N$  чисел. Нужно уметь обрабатывать 3 типа запросов:

- `get(L, R, x)` — сказать, сколько элементов отрезка массива  $[L..R]$  не меньше  $x$ .
- `set(L, R, x)` — присвоить всем элементам массива на отрезке  $[L..R]$  значение  $x$ .
- `reverse(L, R)` — перевернуть отрезок массива  $[L..R]$ .

### Формат входных данных

Число  $N$  ( $1 \leq N \leq 5 \cdot 10^5$ ) и массив из  $N$  чисел. Далее число запросов  $M$  ( $1 \leq M \leq 5 \cdot 10^5$ ) и  $M$  запросов. Формат описания запросов предлагается понять из примера. Для всех отрезков верно  $1 \leq L \leq R \leq N$ . Исходные числа в массиве и числа  $x$  в запросах — целые от 0 до  $10^9$ .

### Формат выходных данных

Для каждого запроса типа `get` нужно вывести ответ.

### Пример

стандартный ввод	стандартный вывод
5	3
1 2 3 4 5	1
6	3
get 1 5 3	1
set 2 4 2	
get 1 5 3	
reverse 1 2	
get 2 5 2	
get 1 1 2	