

Задача А. Луч смерти

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

Уэнсдей очень любит играть с игрушками в своей комнате, особенно ей нравится испепелять их лучом смерти. Представим её комнату в виде декартовой плоскости, тогда положение игрушки в комнате можно описать целочисленными координатами $(x; y)$. В результате своей игры она последовательно ставит или убирает игрушки n раз, то есть совершает изменение одно из двух типов:

- $+ x y$ — поставить игрушку в точку $(x; y)$;
- $- x y$ — убрать игрушку из точки $(x; y)$.

Луч смерти Уэнсдей проходит по какой-то прямой и испепеляет все игрушки на ней. После каждого изменения Уэнсдей интересно, может ли она уничтожить все игрушки в комнате, направив луч смерти по некоторой прямой.

Формат входных данных

В первой строке вводится единственное число n — количество изменений ($1 \leq n \leq 2 \cdot 10^5$).

В следующих n строках содержатся описания изменений, по одному на каждой строке. Каждая строка начинается или с символа «+», или с символа «-», за которым следуют два целых числа x и y ($|x|, |y| \leq 10^9$). Если строка начинается с символа «+», то Уэнсдей ставит игрушку в точку $(x; y)$. Гарантируется, что в таком случае в заданной точке нет игрушек. Если же строка начинается с символа «-», то Уэнсдей убирает игрушку из точки $(x; y)$. Гарантируется, что в этом случае в заданной точке есть игрушка.

Формат выходных данных

Выведите n строк. Каждая строка: «YES» или «NO» в любом регистре — i -я строка отвечает на вопрос Уэнсдей после i -го изменения.

Пример

стандартный ввод	стандартный вывод
7	YES
+ 0 0	YES
+ 0 1	YES
+ 0 -1	NO
+ 1 1	NO
- 0 1	NO
+ -1 -1	YES
- 0 -1	

Задача В. Пирамида Видений

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

Во время расследования таинственных событий в Академии Невермор Уэнсдей находит древний дневник бывшей директрисы Лариссы Уимс. В нём содержится описание магического ритуала, в котором используется Пирамида Видений — древний артефакт, связанный с историей Проклятых и способный дать Уэнсдей подсказку в её расследовании.

Пирамида Видений представляет собой загадочный объект, состоящий из n уровней. На i -м уровне находятся i магических кристаллов, каждый кристалл содержит внутри себя запас тёмной энергии, выражаемый целым числом.

Ритуал призван разрушить данную пирамиду и происходит он по следующим правилам:

- Уровни разрушаются сверху вниз: сначала первый (верхний), затем второй и т.д., пока не будет разрушен предпоследний уровень. Нижний уровень (основание) не разрушается.
- Для разрушения уровня каждый кристалл на нём должен «передать» свою энергию сильнейшему из двух кристаллов под ним. Так как энергия тёмная, у кристалла, получающего энергию, она не увеличивается, а уменьшается на величину переданной ему энергии. Иначе говоря, пусть на некотором уровне стоит кристалл со значением x , а непосредственно под ним на следующем уровне находятся два кристалла со значениями a (левый) и b (правый), тогда:
 - если $a > b$, то энергия передаётся левому кристаллу, и его значение становится $a - x$;
 - если $a \leq b$, то энергия передаётся правому кристаллу, и его значение становится $b - x$;

После выполнения ритуала кристаллы основания Пирамиды Видений наполнятся необходимым количеством тёмной энергии и покажут пророчество о судьбе Академии и тайне, которую скрывала Ларисса Уимс.

Уэнсдей смогла выполнить ритуал — а сможете ли вы?

Формат входных данных

В первой строке задано целое число n — количество уровней в Пирамиде Видений ($1 \leq n \leq 1000$).

Следующие n строк описывают уровни пирамиды.

В i -й строке через пробел записаны i целых чисел $a_{i,j}$, где j -е число описывает величину тёмной энергии, заключённой в j -м кристалле этой строки ($|a_{i,j}| \leq 10^9$).

Формат выходных данных

Выведите n чисел — величины тёмной энергии кристаллов основания пирамиды после выполнения ритуала.

Примеры

стандартный ввод	стандартный вывод
3 1 2 3 4 5 6	4 3 4
1 1000000000	1000000000
5 7 -5 2 -5 2 2 8 1 -4 8 8 -9 3 4 -2	-5 -9 18 -2 -2

Задача С. Побег от зомби

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

Это интерактивная задача.

Как вы знаете, в Академии Невермор живут зомби, которые ходят по территории академии, представленной в виде декартовой плоскости. Зомби — мертвые создания, поэтому очень медлительные и несообразительные. Передвигаются они только параллельно осям координат, и расстояние d в одном направлении они проходят за d^2 секунд (при этом d должно быть целым). Заметьте, что после прохождения расстояния d зомби должен сменить направление движения, чтобы не сойти с ума.

К примеру, валидный путь зомби может выглядеть так:

$$(0, 0) \rightarrow (2, 0) \rightarrow (2, 1) \rightarrow (-1, 1) \rightarrow (-1, -1) \rightarrow (0, -1)$$

И такой маршрут зомби преодолест за $2^2 + 1^2 + 3^2 + 2^2 + 1^2 = 19$ секунд.

Ясно, что маршрут до одной и той же точки может занять у зомби разное количество времени. К сожалению местных жителей, зомби уже научились преодолевать расстояния за оптимальное время. Тогда за $f(x, y)$ обозначим минимальное время, за которое зомби может добраться до точки $(0, 0)$ из точки (x, y) .

Вы решили проверить свою смекалку и узнать, насколько хорошо вы ориентируетесь на местности. Уэнсдей прекрасно знает характер зомби, и она согласилась играть с вами на таких условиях:

Уэнсдей загадывает точку (x_0, y_0) на плоскости с целыми координатами (в неё она поставит зомби). После этого вы можете задавать ей следующие вопросы:

- «? Δx Δy » — после этого вопроса Уэнсдей передвинет зомби на вектор $(\Delta x, \Delta y)$ на плоскости и сообщит вам, сколько времени вы выиграли, когда передвинули зомби на такой вектор: формально, она сообщит вам $f(x + \Delta x, y + \Delta y) - f(x, y)$, где (x, y) — координаты зомби до этого хода. После этого хода зомби переместится на указанный вами вектор.

Естественно, Уэнсдей согласилась играть с вами в эту игру не просто так: вам нужно угадать точку, загаданную Уэнсдей, иначе она скормит вас зомби.

Протокол взаимодействия

Взаимодействие с интерактором проходит в виде запросов со стороны вашей программы и ответов со стороны интерактора. Вы можете выполнить действие, описанное в условии, не более 100 раз.

Чтобы переместить точку, выведите строку в формате «? Δx Δy », после чего загаданная точка переместится на вектор $(\Delta x, \Delta y)$ ($|\Delta x|, |\Delta y| \leq 2 \cdot 10^9$). Интерактор в ответ выведет на отдельной строке ответ Уэнсдэй.

Чтобы вывести ответ на задачу, выведите строку «! x y », где в качестве x и y должны быть **текущие** координаты точки. Этот вывод не учитывается в количестве запросов. После этого интерактор выведет вердикт — «You win», если ваше предположение верно, и «Your brains will be eaten» в противном случае. Если ваш ответ был неверен, ваше решение получит вердикт WA (Wrong Answer), а интерактор завершится. Во избежание получения некорректных вердиктов, считав информацию о том, что выведенный ответ неверен, ваше решение тоже должно завершиться.

Гарантируется, что исходная точка, загаданная Уэнсдэй, имеет целые координаты, по модулю не превосходящие 10^9 .

Если в какой-то момент ваша программа превышает лимит в 100 запросов, ваша программа завершится с вердиктом WA.

Если в какой-то момент координаты точки, загаданной Уэнсдэй, по модулю превысили $8 \cdot 10^9$, то она ответит вам -1 и вы получите вердикт WA.

Важно: не забывайте после каждой выведенной строки сбрасывать буфер вывода, чтобы интерактор получил ваш запрос. Это можно сделать с помощью `std::cout.flush()` в C++,

`System.out.flush()` в Java и `sys.stdout.flush()` в Python, а также аналогичными командами в других языках. Если ваша программа не сбрасывает буфер вывода, она получит вердикт TL (Time Limit Exceeded) или IL (Idleness Limit Exceeded).

Пример

стандартный ввод	стандартный вывод
? 2 -1	6
? -2 0	2
? -1 1	2
? 1 0	3
! 1 2	You win

Замечание

В первом примере изначально загадана точка $(1, 2)$, и в результате запросов она двигается таким образом:

$$(1, 2) \rightarrow (3, 1) \rightarrow (1, 1) \rightarrow (0, 2) \rightarrow (-1, -1) \rightarrow (1, 2)$$

Задача D. Великая теорема Фестера

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

Уэнсдей Аддамс, ученица Академии Невермор, как обычно проводит свои мрачные эксперименты. Недавно она нашла в старой тетради дяди Фестера странную формулу: $a^n + b^n = c^n$.

Формула показалась ей подозрительно простой, и Уэнсдей решила проверить, существует ли хотя бы одно натуральное решение для разных значений n и c .

Пусть для $n > 2$ решений действительно не существует — рядом с формулой, на полях тетради, дядя Фестер написал удивительно простое доказательство этого факта, и Уэнсдей не собирается оспаривать его. При $n = 2$ это уравнение вызывает у нее иррациональную неприязнь, поэтому такие случаи она рассматривать не будет. Остаются только неотрицательные $n < 2$, для которых она будет проверять, существуют ли у этого уравнения решения.

Ваша задача — помочь Уэнсдей проверить по данным n и c , существует ли такая пара (a, b) , что $a^n + b^n = c^n$.

Стоит помнить, что выражение 0^0 не имеет смысла — если при вычислениях встречается такая ситуация, Уэнсдей считает, что равенство не выполняется.

Формат входных данных

В единственной строке ввода через пробел даны два целых числа n и c — параметры уравнения из формулы дяди Фестера ($0 \leq n \leq 1$; $-10^9 \leq c \leq 10^9$).

Формат выходных данных

Если существуют такие целые a и b от -10^9 до 10^9 , для которых выполняется равенство $a^n + b^n = c^n$, в первой строке выведите «YES» (без кавычек), а затем во второй строке через пробел выведите сами числа a и b .

Если решения соответствующего уравнения нет, в единственной строке выведите «NO».

Обратите внимание, что a и b , удовлетворяющие уравнению, но не лежащие в обозначенных границах ($|a|, |b| \leq 10^9$), приниматься не будут.

Пример

стандартный ввод	стандартный вывод
1 5	YES 2 3

Задача Е. Лабиринт Кошмаров

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

Пытаясь спасти своего друга, Уэнсдей оказывается в Лабиринте Кошмаров, созданном древним проклятием. Структура лабиринта представляет собой дерево из n комнат и $n - 1$ коридоров между ними. Каждая комната имеет уникальный номер от 1 до n .

Проклятие, наложенное на лабиринт, искажает пространство так, что Уэнсдей способна из своей текущей комнаты переместиться не только в соседние комнаты, но и все комнаты, до которых можно пройти, пройдя не более трёх коридоров. Иначе говоря, если кратчайший путь между двумя комнатами в лабиринте состоит не более чем из трёх коридоров, то Уэнсдей способна сразу перемещаться между этими комнатами, минуя другие.

Чтобы снять проклятие и найти выход, Уэнсдей должна, начав в любой комнате, обойти все комнаты ровно по одному разу и вернуться в начальную. Выведите любой возможный маршрут нашей мрачной героини.

Формат входных данных

В первой строке задано целое число n — количество комнат в Лабиринте Кошмаров ($3 \leq n \leq 2 \cdot 10^5$).

Каждая из следующих $n - 1$ строк содержит пару целых чисел u и v , означающую, что между комнатами u и v существует коридор ($1 \leq u, v \leq 2 \cdot 10^5$).

Гарантируется, что лабиринт является деревом.

Формат выходных данных

Выведите последовательность комнат — маршрут Уэнсдей.

Примеры

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

Задача F. Ларёк и две куклы

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

В поздний бледный вечер девочка Уэнсдей ~~с последней парты~~ прогуливалась по кладбищенской аллее и наткнулась на странный ларёк, загадочный хозяин которого сделал ей странное предложение. Как известно, Уэнсдей всегда имеет при себе две куклы Вуду: деревянную и тряпичную. Деревянная заключает в себе a проклятий, тряпичная — b проклятий. Владелец ларька способен с помощью редкой тёмной магии одновременно изменить куклы следующим образом:

- количество проклятий в деревянной кукле изменяется с x на $(x + c) \bmod m$;
- количество проклятий в тряпичной кукле изменяется с y на $(y - c) \bmod m$.

Уэнсдей может попросить применить заклинание любое число раз (включая ноль), но только одновременно к обеим куклам.

Естественно, Уэнсдей хочется, чтобы её куклы содержали как можно больше проклятий, но она опасается, что в предложении хозяина ларька кроется подвох, а потому ей хочется узнать: какую максимальную итоговую сумму проклятий в двух куклах она сможет получить, если примет его предложение?

Помогите Уэнсдей посчитать эту величину, прежде чем ларёк закроется на ночь.

Формат входных данных

В первой и единственной строке содержится 4 целых числа a, b, c, m — первые два числа обозначают изначальное количество проклятий в деревянной и тряпичной куклах соответственно, следующие два числа — параметры, по которым по которым изменяются куклы ($0 \leq a, b, c < m \leq 10^9$).

Формат выходных данных

Выведите единственное число — ответ на задачу.

Пример

стандартный ввод	стандартный вывод
30 47 1 80	157

Задача G. Подарок Уэнсдей

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	5 секунд
Ограничение по памяти:	256 мегабайт

На свой день рождения Уэнсдей получила длинную строку s . Как известно, Уэнсдей ненавидит подарки, не связанные с пытками, поэтому увидев такой ужасный подарок, она сразу же захотела покрасовать его на мелкие части. Однако перед тем как приступить к своей затее, она заметила, что строку s можно разрезать так, что в результате получится собрать две проклятые строки t_1 и t_2 , которые очень нужны ей для пыток.

Уэнсдей может разрезать строку s несколько раз в любом месте. После этого она возьмет изначально пустые строки a и b и рассмотрит получившиеся после разрезания части s слева направо: каждую часть она припишет либо в конец строки a , либо в конец строки b . В конце должно получиться:

$$a = t_1, \quad b = t_2, \quad |a| + |b| = |s|.$$

Уэнсдей не терпится заполучить проклятые строки и приступить к пыткам, поэтому она просит Вас определить минимальное количество разрезов строки s , достаточное, чтобы описанным способом можно было получить строки t_1 и t_2 .

Формат входных данных

Первая строка входных данных содержит строку s ($3 \leq |s| \leq 10^4$).

В следующих двух строках входных данных содержатся непустые строки t_1 и t_2 соответственно.

Гарантируется, что $|t_1| + |t_2| = |s|$. Все строки состоят из строчных латинских букв.

Формат выходных данных

Выведите единственное число — минимальное количество разрезов строки s , чтобы можно было получить строки t_1 и t_2 описанным выше способом. Гарантируется, что ответ всегда существует.

Примеры

стандартный ввод	стандартный вывод
wednesenidday wednesday enid	2
shadowwednesdayw wednesday shadow	2

Замечание

Пример 1:

Можно разрезать строку следующим образом: `wednes | enid | day`

Тогда первая и последняя часть образуют строку t_1 , а средняя — строку t_2 .

Пример 2:

Можно разрезать строку следующим образом: `shado | wednesday | w`

Тогда первая и последняя часть образуют строку t_2 , а средняя — строку t_1 .

Задача Н. Ужасающий эксперимент Уэнсдей

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

В рамках своего последнего мрачного эксперимента Уэнсдей решила использовать таинственные числовые последовательности. Она обнаружила, что некоторые комбинации чисел содержат скрытую гармонию тьмы, выраженную через сумму трех мистических дробей. Её задача — найти среди массива чисел ту единственную тройку, чья общая энергия будет минимальной, ведь слишком большой объём энергии мгновенно убивает, а жертва должна долго мучаться перед смертью.

Вам предстоит повторить этот эксперимент: найдите три различных индекса i, j, k таких, чтобы сумма

$$\frac{a_i}{a_k} + \frac{a_j}{a_i} + \frac{a_k}{a_j}$$

была минимальна. В качестве ответа выведите минимальное значение этой суммы.

Помните — как и в любом деле, связанном с тёмной энергией, малейшая неточность ведёт к ужасным последствиям, а потому ваш ответ должен отличаться от истинного не более чем на 10^{-9} .

Формат входных данных

В первой строке вводится число n ($3 \leq n \leq 2 \cdot 10^5$).

В следующей строке вводится n чисел a_i ($1 \leq a_i \leq 10^9$).

Формат выходных данных

В единственной строке выведите ответ на задачу.

Ваш ответ будет считаться верным, если абсолютно отличается от правильного не более чем на 10^{-9} . Другими словами, если ваш ответ равен ans_p , а правильный ответ равен ans_j , то он считается верным, если $|\text{ans}_p - \text{ans}_j| \leq 10^{-9}$.

Пример

стандартный ввод	стандартный вывод
4 3 6 2 4	3.333333333

Задача I. Проблемы бритья

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

Гомес Аддамс очень любит свои усы, однако за ними надо ухаживать, поэтому каждое утро он берёт свою бритву, сбрасывает щетину и подравнивает свои усы.

Лицо Гомеса можно представить как матрицу размера $n \times m$, где каждая клетка обозначает какую-то область на лице. На этой матрице символами «+» обозначим области, которые Гомес хочет побрить, символами «#» области, которые Гомес не хочет брить, а символами «.» области, на которые ему безразлично, брить там или нет. У Гомеса есть бритва ширины k (которую можно поворачивать на 90 градусов), однако бритва не бреет, если её просто приложить — её нужно вести по направлению лезвий, а значит, Гомес может за раз побрить область размера $k \times l$ (или $l \times k$), где $l > 1$. Обратите внимание, что бритва не может выходить за границы поля.

Гомес Аддамс хочет узнать, сможет ли он побрить все необходимые области на своём лице с помощью имеющейся у него бритвы, или же он снова будет ходить недовольный состоянием своих усов.

Формат входных данных

В первой строке содержатся числа n, m, k — размеры лица Гомеса и размер бритвы соответственно ($1 \leq n, m, k \leq 10^5$; $1 \leq n \times m \leq 101\,000$).

Далее идут n строк, каждая из которых содержит m символов из множества {«+», «#», «.»} и описывает состояние лица Гомеса.

Формат выходных данных

Выведите «YES», если Гомес сможет побрить все необходимые области, и «NO» в противном случае.

Примеры

стандартный ввод	стандартный вывод
9 10 3 ##### ###...#####..##.. +.....+ ++.++++.++ +++##### +++++++ +++++++	YES
9 10 3 ##### ###...### ..++..++.. ..##..##.. +.....+ ++.++++.++ +++##### +++++++ +++++++	NO

Задача J. LOIS

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

Подобравшись к разгадке того, что такое LOIS, Уэнсдей попала в комнату с единственной дверью. На двери был написан массив, состоящий из целых чисел от 0 до k включительно. Как подсказал ей дядя Фестер, для открытия двери нужно было придумать другой (не равный данному) массив, содержащий числа от 0 до k включительно, «похожий» на него.

Как известно, дядя Фестер считает два массива чисел «похожими», если сумма чисел в первом массиве равна сумме чисел во втором массиве, и произведение чисел в первом массиве равно произведению чисел во втором массиве.

Требуется найти массив, не равный исходному, но похожий на него, или сообщить, что это невозможно.

Формат входных данных

В первой строке содержатся два целых числа n и k — длина массива, записанного на двери, и верхняя граница значений ($1 \leq n, k \leq 10^5$).

Во второй строке содержатся n целых чисел $a_1, \dots, a_n$ — элементы массива ($0 \leq a_i \leq k$).

Формат выходных данных

Выведите -1 , если такого массива нет, и Уэнсдэй не сможет открыть дверь.

Иначе в первой строке выведите m — длину нужного для разгадки массива ($1 \leq m \leq 3 \cdot 10^5$).

Во второй строке выведите m чисел от 0 до k — массив, который приводит к разгадке.

Примеры

стандартный ввод	стандартный вывод
5 6 1 2 3 4 5	6 5 3 2 2 2 1
3 2 0 0 0	1 0
4 7 1 1 1 1	-1

Задача К. Инциденты в Неверморе

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	4.5 секунд
Ограничение по памяти:	512 мегабайт

В Академии Невермор по длинному коридору (его можно считать числовой прямой) стоят n обитателей. У i -го обитателя есть:

- координата на коридоре x_i ;
- личный запас ума m_i ;
- показатель командной работы z_i ;
- особый трюк: он может временно уменьшить свой ум и повисить командную работу.

Формально, у i -го обитателя есть два числа c_i и a_i . В любой момент (за нулевое время) он может заменить своё текущее значение ума на

$$\left\lfloor \frac{\text{текущий ум}}{c_i} \right\rfloor$$

и одновременно увеличить свою командную работу на a_i . Этой операцией можно пользоваться произвольное количество раз, если это выгодно.

Обитатели могут перемещаться по коридору: за единицу времени каждый обитатель может сдвинуться по прямой на $+1$ или -1 .

Если несколько обитателей в некоторый момент времени оказались в одной и той же точке коридора, они могут объединиться в команду. Тогда:

- ум команды равен сумме умов всех участвующих обитателей;
- командная работа команды равна минимуму по командной работе всех участвующих обитателей.

В академии возникают q независимых инцидентов. Для каждого инцидента известны:

- координата y_j — место, где нужно собрать команду;
- требуемый ум s_j ;
- требуемая командная работа t_j .

Необходимо определить, за какое минимальное время в точке y_j можно собрать такую команду из имеющихся обитателей (с учётом их перемещений и применения трюков), чтобы суммарный ум команды был не меньше s_j , а командная работа команды не меньше t_j . Если собрать такую команду невозможно, следует вывести -1 .

Формат входных данных

В первой строке каждого набора входных данных содержатся два целых числа n, q — количество обитателей и количество инцидентов ($1 \leq n, q \leq 10^5$).

Во второй строке содержатся n целых чисел — x -координаты обитателей ($0 \leq x_i \leq 10^9$).

В третьей строке содержатся n целых чисел m_i ($0 \leq m_i \leq 10^5$).

В четвёртой строке содержатся n целых чисел z_i ($0 \leq z_i \leq 10^5$).

В пятой строке содержатся n целых чисел c_i ($1 \leq c_i \leq 10^5$).

В шестой строке содержатся n целых чисел a_i ($1 \leq a_i \leq 10^5$).

Далее следуют q строк, в каждой из которых содержатся три целых числа y_j, s_j, t_j — координата инцидента, требуемый ум и требуемая командная работа соответственно ($0 \leq y_j \leq 10^9$; $0 \leq s_j, t_j \leq 10^{14}$).

Формат выходных данных

Для каждого из q инцидентов выведите одно целое число — минимальное время, за которое в точке y_j может быть собрана команда с суммарным умом не меньше s_j и командной работой не меньше t_j . Если это невозможно, выведите -1 .

Пример

стандартный ввод	стандартный вывод
4 6	5
0 10 5 7	10
10 6 4 3	3
1 5 2 4	-1
2 2 2 3	-1
3 1 2 1	2
5 12 4	
0 6 6	
10 7 4	
5 25 3	
5 15 5	
7 5 4	

Задача L. Мистический эксперимент Пагсли

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

Стараясь не отставать от своей сестры, Пагсли тоже решил провести в одной из тёмных лабораторий особняка Аддамс странный эксперимент. На этот раз он собрал массив из n чисел, в каждом из которых, по его словам, запечатана *частица семейного безумия*.

В ходе опытов Пагсли обнаружил нечто, что он назвал равновесием хаоса: некоторые элементы массива обладают загадочной гармонией — если посчитать **xor** всех элементов слева от некоторого элемента, а затем **xor** всех элементов справа, оба значения окажутся одинаковыми. Такие элементы массива он назвал точками равновесия хаоса. Напомним, что **xor** — это операция побитового исключающего или, а также **xor** пустого префикса или суффикса равен 0.

Пагсли уверен, что именно такие элементы массива удерживают особняк Аддамс от разрушения. Теперь он хочет уметь быстро узнавать, сколько таких элементов осталось, ведь эксперименты часто заканчиваются непредсказуемыми взрывами, из-за чего числа в массиве постоянно меняются!

Пагсли, конечно, талантлив, но до сестры ему ещё далеко, поэтому он просит вас написать программу, которая сможет обрабатывать его запросы:

- 1 i x — заменить элемент в i -й позиции (считая с нуля) на значение x . Гарантируется, что элемент в массиве с таким индексом существует;
- 2 — получить информацию о том, сколько точек равновесия хаоса существует в момент запроса.

Помогите Пагсли, ведь в противном случае равновесие может нарушиться и ничто не сможет спасти особняк от разрушения!

Формат входных данных

В первой строке содержится целое число n — количество чисел в массиве ($1 \leq n \leq 2 \cdot 10^5$).

Вторая строка содержит последовательность из n целых чисел $a_1, a_2, \dots, a_n$ — элементы данного массива ($0 \leq a_i \leq 10^9$).

Третья строка содержит целое число q — количество запросов ($1 \leq q \leq 2 \cdot 10^5$).

В следующих q строках содержатся запросы, по одному на каждой строке. Каждая строка начинается с целого числа k — типа запроса ($1 \leq k \leq 2$). Если тип запроса равен 1, то далее следует 2 целых числа i, v — индекс элемента массива, который нужно заменить, а также значение, на которое нужно заменить ($0 \leq i < n, 0 \leq v \leq 10^9$).

Формат выходных данных

На каждый запрос второго типа выведите в отдельной строке единственное число — количество позиций, для которых **xor** префикса равен **xor** суффикса в текущем массиве.

Пример

стандартный ввод	стандартный вывод
10 0 1 2 3 4 5 6 7 8 13 2 1 9 9 2	1