

Задача А. Бесплатная и платная операции

Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Перед вами массив из n нулей, массив индексируется с единицы. Вы хотите превратить его в массив a . Для этого вы сначала можете один раз выбрать некоторое целое число x и выполнить присвоение $a_i = 1$ для всех $1 \leq i \leq x$ (при желании вы можете выбрать $x = 0$ и не менять массив). Это изменение не стоит ничего.

Затем вы можете несколько раз применить другую операцию, каждое применение стоит одну монету. Вы выбираете целое число y и выполняете присвоение $a_y = 1 - a_y$. Какое минимальное количество монет вам потребуется, чтобы превратить массив нулей в массив a ?

Формат входных данных

В первой строке дано целое число n ($1 \leq n \leq 2 \cdot 10^5$). Следующая строка содержит массив целых чисел a ($0 \leq a_i \leq 1$).

Формат выходных данных

Выведите одно число — ответ на задачу.

Примеры

стандартный ввод	стандартный вывод
5 0 0 0 0 1	1
7 1 1 0 1 1 0 1	2

Замечание

В первом примере $x = 0, y = 5$. Во втором примере $x = 7, y = 6, y = 3$.

Задача В. Круглая парковка и пачки машин

Ограничение по времени: 0.5 секунд
Ограничение по памяти: 64 мегабайта

Имеется n парковочных мест, расположенных в круг и пронумерованных от 0 до $n-1$. На парковку приезжают машины, у каждой машины есть предпочтительное место x . Если предпочтительное место свободно, машина паркуется на нём и занимает его. Иначе она продолжает движение по кругу к месту $x+1$, пока не найдет свободное место. Когда машина проезжает место $n-1$, она подъезжает к месту 0.

По списку машин определите минимальный номер свободного места на парковке. Можно показать, что он не зависит от порядка приезда машин. Предпочтительные места машин описываются особым образом, прочитайте формат входных данных.

Формат входных данных

В первой строке даны два целых числа n и k ($2 \leq n \leq 3 \cdot 10^6$, $1 \leq k \leq 10^4$). Число k задает количество строк описаний желаемых парковочных мест группы машин.

Каждая из следующих k строк содержит четыре целых числа x, y, a, b ($0 \leq a, b \leq 10^9$), обозначающих, что есть по x машин, предпочитающих парковочное место $f(1), f(2), \dots, f(y)$, где $f(i) = (ai + b) \% n$. Таким образом описывается xy машин. Гарантируется, что $\sum xy \leq n - 1$.

Формат выходных данных

Выведите одно число — ответ на задачу.

Пример

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

Замечание

В примере одна машина хочет место $(1 \cdot 1 + 7) \bmod 10 = 8$, две машины хотят место $(0 \cdot 1 + 1) \bmod 10 = 1$ (займут 1 и 2), три машины хотят место $(2 \cdot 1 + 4) \bmod 10 = 6$ (займут 6, 7 и 9), и ещё три машины хотят место $(2 \cdot 2 + 4) \bmod 10 = 8$ (займут 0, 3, 4).

Задача С. Привет от Снарка

Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

Все вы знаете о числовой системе древних римлян. В самом строгом виде она определяется следующим образом:

1. Различные символы (латинские буквы) обозначают значения $1, 5, 10, 50, \dots, 1 \cdot 10^i, 5 \cdot 10^i, 1 \cdot 10^{i+1}, 5 \cdot 10^{i+1}, \dots$;
2. Римское число записывается как последовательность символов, сумма значений этих символов дает значение самого числа, кроме случаев, когда применяется правило вычитания (пункты 3 и 4);
3. Правило вычитания гласит, что символ со значением вида $1 \cdot 10^i$ может быть помещен перед символом со значением $5 \cdot 10^i$ или $1 \cdot 10^{i+1}$, тогда он участвует в сумме с отрицательным знаком;
4. Правило вычитания может применяться только один раз для каждого i ;
5. В римском числе символы упорядочены в неубывающем порядке их значений (кроме случаев применения вышеупомянутого правила);
6. Римские числа записываются с минимальным количеством символов.

Примеры валидных римских чисел: I, III, IV, VIII, IX, XVIII, XLII. Примеры невалидных римских чисел: IIII (пункт 6), VX (пункты 3, 6), IC (пункт 3), IXIX и IXX (пункт 4), IVX (пункты 3, 4), XVIIC (пункт 5).

Символы в классической римской системе счисления: I = 1, V = 5, X = 10, L = 50, C = 100, D = 500, M = 1000. Наибольшее число, которое можно записать в этой системе, равно MMMCMXCIX = 3999. Очевидно, римлянам не приходилось работать с большими числами. Поэтому мы расширим систему следующими символами: Q, i, v, x, l, c, d, m, q, a, A, b, B, e, E, f, F, g, G, h, H, j, J, k, K, n, N, o, O, p, P, r, R, s, S, t, T, u, U, w, W, y, Y, z и Z, где Q = 5000, i = 10000, v = 50000, x = 100000 и так далее.

Например, в такой расширенной системе число $2^{31} - 1$ записывается как *II^{XXXXI} - I = bbamqdccxlviiiiMMMDCLVII*.

Сева и Денис общаются по зашифрованному коммуникационному каналу с использованием шифра. Подслушивающий их Тихон узнал, что каждый день они меняют шифровальный ключ. Сам ключ они передают по незащищенному каналу и для запутывания противника записывают его как расширенное римское число. Для дополнительного замешательства в сообщении есть и другие символы помимо латинских букв: арабские цифры, знаки препинания и т.п. (разрешены все ASCII символы с кодами от 33 до 126 включительно).

Тихон записал одно такое сообщение длиной n и хочет извлечь из него самое длинное валидное расширенное римское число, которое является его подпоследовательностью (символы в нем идут в том же порядке, но не обязательно рядом). Перестановки символов не разрешены. В случае если из сообщения можно извлечь более одного валидного римского числа максимальной длины, ее интересует самое большое из них. Помогите ему определить это число.

Формат входных данных

На первой строке дано положительное число n ($I \leq n \leq X^V$) — длина сообщения. На второй строке будет дано само сообщение, состоящее из n символов. Гарантируется, что сообщение будет содержать хотя бы одну латинскую букву.

Формат выходных данных

Выведите искомое валидное расширенное римское число.

Примеры

стандартный ввод	стандартный вывод
12 (XVII) < (MCM)	XVII
11 (XVI) < (MCM)	MCM
12 (MCM) < (XVII)	MCMXVII

Задача D. Без AC

Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Дано положительное четное число n . Необходимо найти количество строк s длины n , состоящих из символов A, B и C , которые можно преобразовать в пустую строку, повторяя следующую операцию:

Выберем два соседних символа в s и удалим их. Однако выбирать пары AC или CA не разрешается. Например, в строке $ACCB$ мы можем удалить CC , затем AB .

Ответ может быть огромным, поэтому вычислите количество строк по модулю 998244353.

Формат входных данных

Дано одно целое четное n ($2 \leq n \leq 10^7$).

Формат выходных данных

Выведите одно число — ответ на задачу.

Примеры

стандартный ввод	стандартный вывод
2	7
4	63

Задача Е. Игра гномов

Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Враг и его армия приближаются к вашей крепости, и все, что у вас есть для защиты, – это легион охранных гномов. Нет надежды на победу в битве, поэтому ваш фокус будет сосредоточен на том, чтобы нанести как можно больше урона врагу.

У вас есть n гномов. Перед битвой их необходимо разделить на максимум m непустых групп. Битва будет проходить по раундам. В каждом раунде ваши гномы будут атаковать врага, нанося одну единицу урона за каждого живого гнома. Затем враг будет атаковать, бросая молнию в одну из m групп. Молния убивает k гномов в этой группе или всех, если количество живых гномов в группе меньше k . Битва заканчивается, когда все гномы мертвы. Враг всегда будет бросать молнии оптимальным образом, чтобы минимизировать общий урон, причиняемый гномами.

Теперь вы задаетесь вопросом, какой максимальный урон вы можете нанести врагу, если разделите гномов на группы оптимальным образом?

Формат входных данных

Даны три целых числа n, m, k ($1 \leq n \leq 10^9, 1 \leq m, k \leq 10^7$).

Формат выходных данных

Выведите одно число — ответ на задачу.

Пример

стандартный ввод	стандартный вывод
10 4 3	27

Замечание

Например, в примере у вас 10 гномов, которые должны быть разделены на не более чем 4 группы, и молния наносит максимум 3 урона, то оптимальным решением будет создать одну большую группу размером 7 и три маленькие группы размером 1. В первом раунде вы наносите 10 урона, и молния уменьшает большую группу на 3. В следующем раунде вы наносите 7 урона, и большая группа уменьшается до размера 1. В оставшиеся четыре раунда вы наносите 4, 3, 2 и 1 урона соответственно, и молния уничтожает одну группу в каждом раунде. В итоге вы наносите $10 + 7 + 4 + 3 + 2 + 1 = 27$ урона.

Задача F. Разложить строку на две одинаковые

Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Дана строка s длины $2n$ из строчных английских букв. Определите, можно ли выбрать подпоследовательность ровно из n букв таким образом, чтобы строки, сформированные выбранными буквами и невыбранными буквами, были одинаковыми. Порядок букв менять нельзя, каждую букву можно либо выбрать, либо не выбрать.

Формат входных данных

Дана строка s , состоящая из строчных английских букв ($2 \leq |s| \leq 40$), длина строки четная.

Формат выходных данных

Выведите Yes, если можно выбрать половину букв требуемым образом, иначе выведите No.

Примеры

стандартный ввод	стандартный вывод
aabb	Yes
abba	No

Задача G. Экотропы, но уже без легенды

Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Дан взвешенный неориентированный граф, состоящий из n вершин и m ребер. Выберем некоторое подмножество из k вершин и назовем эти вершины важными (а остальные неважными). Вы должны выбрать подмножество из $n - 1$ ребер таким образом, чтобы выбранный подграф был деревом, и ровно w ребер соединяли важную вершину с неважной. При этом суммарный вес выбранных ребер должен быть минимально возможным.

Формат входных данных

Первая строка содержит четыре целых числа n, m, k и w , где n ($1 \leq k, w < n \leq 2 \cdot 10^5$, $1 \leq m \leq 5 \cdot 10^5$).

Каждая из следующих k строк содержит одно целое число s ($1 \leq s \leq n$), задающее номер важной вершины. Эти значения будут уникальными и отсортированными по возрастанию. Каждая из следующих m строк состоит из трех целых чисел u, v и c , описывающих ребро между u и v ($1 \leq u \neq v \leq n$) с весом c ($1 \leq c \leq 10^5$). Кратных ребер нет.

Формат выходных данных

Выведите одно целое число — минимальный суммарный вес выбранных ребер, или -1 , если это невозможно.

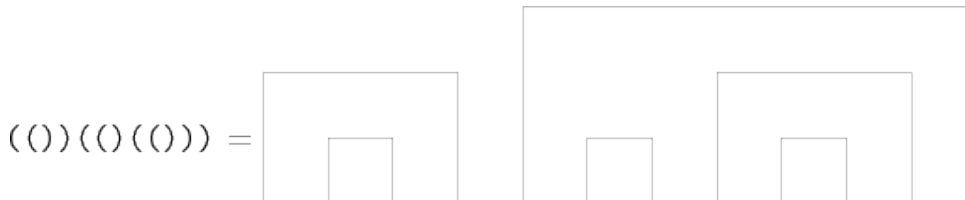
Примеры

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

Задача Н. ПСП в прямоугольники

Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Демид готовит слайды для своей лекции по разбору арифметических выражений. В первой части лекции он хочет сосредоточиться только на разборе скобок. Он придумал интересное геометрическое представление правильной скобочной последовательности для своих студентов, потому что одно изображение лучше тысячи слов.



Формально, определение геометрического представления выглядит следующим образом. Самая простая правильная последовательность скобок $()$ представляется квадратом 1×1 . Если S — это правильная последовательность скобок, а $g(s)$ — её представление, то представление для (S) — это $g(S)$, окруженное прямоугольником, который на две единицы шире, чем $g(S)$, и на одну единицу выше самой высокой точки $g(S)$. Если S и T — две правильные скобочные последовательности, а $g(S)$ и $g(T)$ их представления, то мы получаем $g(ST)$, разместив $g(T)$ на одну единицу правее от $g(S)$.

После того как Демид закончил свои слайды, он начал играть с подготовленными изображениями. Он чередовал окраску ограниченных областей изображений в черный и белый цвета так, что внешняя (неограниченная) область была окрашена в белый цвет. Вам нужно по правильной скобочной последовательности определить, какова суммарная площадь черных фигур.



Формат входных данных

В первой строке дано число T ($1 \leq T \leq 10$) — количество тестовых случаев, для которых нужно решить задачу. Следующие T строк содержат по одной правильной скобочной последовательности из круглых скобок, длина каждой из этих строк не превышает $4 \cdot 10^5$.

Формат выходных данных

Для каждого тестового случая выведите ответ в том порядке, в котором они даны на вход.

Пример

стандартный ввод	стандартный вывод
1 ()()()	20

Замечание

Правильная скобочная последовательность (ПСП) — это строка из открывающих и закрывающих скобок, которая получается из арифметических выражений удалением всего, кроме скобок.

Задача I. Кратчайшая программа для робота

Ограничение по времени: 5 секунд
Ограничение по памяти: 256 мегабайт

У вас есть лабиринт на клетчатой прямоугольной доске размера $N \times M$. В одной из клеток доски имеется робот, который занимает одну клетку и может выполнять программу, каждая команда которой это одна из пяти команд:

- N – переместиться на клетку вверх
- S – переместиться на клетку вниз
- W – переместиться на клетку влево
- E – переместиться на клетку вправо
- G(i) – прыгнуть на i -ю команду программы (индексация с единицы)

После выполнения команды первых четырех типов робот переходит к следующей строке программы или застывает навечно, если её не существует. Строка с номером i при прыжке должна существовать. Например, если робот выполнит программу

1. N
2. E
3. G(6)
4. S
5. G(1)
6. W
7. G(4)

он переместится вверх, вправо, перепрыгнет на строку 6 без перемещения, влево, затем на строку 4, вниз, и всё сначала.

Некоторые клетки доски сломаны, так что если робот заедет на такую клетку, он развалится. Аналогично, робот не может выезжать за границы доски. Задача робота – оказаться в заданной клетке. Как только робот оказывается там, считается, что робот выполнил свою миссию успешно, даже если есть еще невыполненные команды в программе. Ваша задача определить, какое наименьшее число строк должно быть в программе, чтобы робот выполнил свою миссию.

Формат входных данных

Первая строка входных данных содержит T ($1 \leq T \leq 100$) — количество тестовых случаев. Далее следуют описания T тестовых случаев. Каждый тестовый случай начинается со строки, содержащей N и M ($2 \leq N, M \leq 100$) — количество строк и столбцов доски. Затем следуют N строк, каждая из которых содержит строку из M символов. j -й символ i -й строки A_{ij} представляет собой клетку в i -й строке и j -м столбце доски. Символ равен #, если клетка сломана, заглавной S, если клетка содержит робота, заглавной F, если клетка является финишной клеткой, и точке, если клетка не занята и не сломана.

Гарантируется, что $\sum NM \leq 10^4$ по всем тестовым случаям одного запуска.

Формат выходных данных

Для каждого тестового случая выведите одну строку, содержащую одно число — минимальное число строк в программе, если можно составить программу, или No, если это невозможно.

Пример

стандартный ввод	стандартный вывод
5	6
5 5	7
..F##	No
###	5
...#	4
##.##	
##..S	
2 5	
F#...	
...#S	
3 3	
#S#	
###	
#F#	
5 5	
F..##	
###	
...#	
##.##	
##..S	
2 5	
F...#	
....S	

Задача J. Пары карточек

Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

В вашем распоряжении есть карточки с числами, у вас всего n номиналов, a_i карточек с числом i . Вы можете взять две карточки разных номиналов и отложить их, сформировав пару. Требуется сформировать как можно больше пар. При этом, если существует несколько способов сделать это, нужно лексикографически минимизировать набор этих пар. То есть у вас должно быть как можно больше пар $(1, 2)$, затем как можно больше пар $(1, 3)$, затем $(1, n)$, затем $(2, 3)$ и так далее.

Более формально, как сравнить два набора пар. Упорядочим карточки с числами в паре по возрастанию, а после этого упорядочим пары по возрастанию первого элемента, а при равенстве по возрастанию второго элемента. Тогда, если i -я это первая пара, которая отличается в двух наборах, то лексикографически меньше тот набор, в котором i -я пара меньше.

Формат входных данных

В первой строке дано целое число n ($2 \leq n \leq 10^5$). Вторая строка содержит n целых чисел: a_1, a_2, \dots, a_n ($1 \leq a_i \leq 10^9$).

Формат выходных данных

Первая строка должна содержать m — количество различных типов пар, которые нужно сформировать. Следующие m строк должны содержать по три целых числа: i, j, c_{ij} — номиналы карточек, которые нужно объединить в пару и количество таких пар. В каждом описании должно выполняться $i < j$, выводите пары в порядке возрастания i , при равном i в порядке возрастания j .

Пример

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

Задача К. Ожерелье

Ограничение по времени: 1 секунда
Ограничение по памяти: 128 мегабайт

Дано ожерелье, состоящее из n бусин, где каждая бусина принадлежит к одному из k возможных видов. Бусины пронумерованы целыми числами от 1 до n . Бусина под номером i соседствует с бусинами под номерами $i + 1$ и $i - 1$; кроме того, бусины под номерами 1 и n также соседствуют.

Мы хотели бы разделить ожерелье двумя разрезами на две непустые части так, чтобы каждый вид бусин встречался ровно в одной части (т.е. если в одной части есть бусина вида j , то в другой не должно быть бусины этого вида). Сколько таких разбиений существует и какова минимальная разница в длинах двух получившихся частей?

Формат входных данных

Первая строка содержит два целых числа n и k ($2 \leq k \leq n \leq 10^6$). Вторая строка ввода содержит последовательность из n целых чисел r_1, r_2, \dots, r_n ($1 \leq r_i \leq k$), разделенных одинарными пробелами; число r_i определяет вид бусины i . Каждый вид бусины встречается в ожерелье.

Формат выходных данных

Выведите два числа на одной строке. Первое из этих чисел должно быть числом допустимых разделов ожерелья (гарантируется, что для входных данных существует хотя бы один). Второе число должно быть минимальной разницей в длинах полученных частей.

Пример

стандартный ввод	стандартный вывод
9 5 1 4 2 1 1 3 5 5 2	4 3

Замечание

В примере возможны следующие четыре варианта видов бусин для короткой части: (4), (3), (5, 5), (3, 5, 5).

Задача L. Прямые и вложенные круги

Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Даны n прямых на плоскости. Каждая прямая задана двумя различными точками, которые она соединяет. Вам поступают q запросов, каждый запрос является кругом. Вам нужно определить, сколько прямых имеют с очередным кругом хотя бы одну общую точку. Интерес задачи в том, что гарантируется, что круг i целиком вложен в круг $i + 1$ (даже касание исключено).

Формат входных данных

В первой строке дано целое число n ($1 \leq n \leq 10^5$). Следующие n строк задают прямую с помощью двух различных целочисленных точек x_1, y_1, x_2, y_2 ($-10^8 \leq x_1, y_1, x_2, y_2 \leq 10^8$). Далее идет целое число q ($1 \leq q \leq 10^5$). Следующие q строк задают круг с помощью трёх целых чисел: координат центра x, y и радиуса r ($-10^8 \leq x, y \leq 10^8, 1 \leq r \leq 10^8$). Гарантируется, что ответ не изменится, даже если изменить радиус круга в любую сторону на $r \cdot 10^{-12}$.

Формат выходных данных

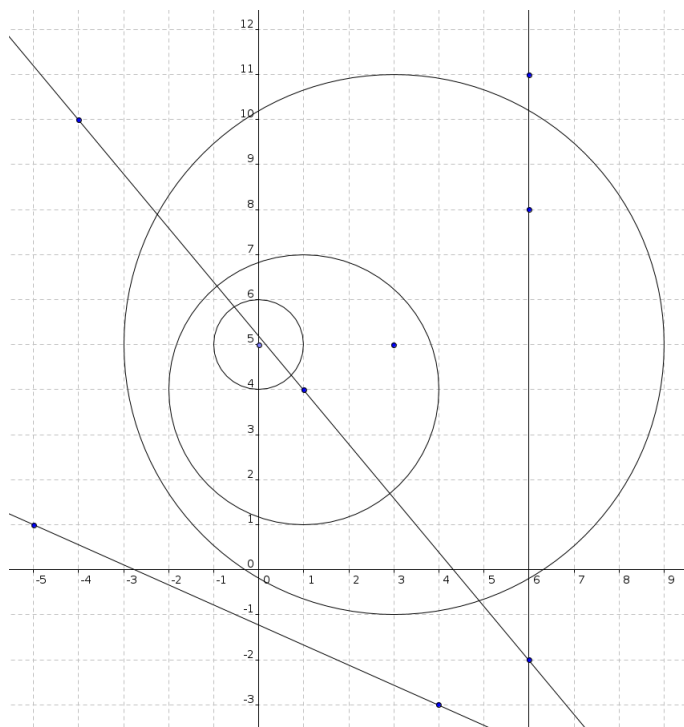
В i -й строке выведите число прямых, пересекающихся с i -м кругом.

Пример

стандартный ввод	стандартный вывод
4	1
6 8 6 11	1
-4 10 6 -2	3
6 11 6 8	
-5 1 4 -3	
3	
0 5 1	
1 4 3	
3 5 6	

Замечание

На картинке изображен пример



Задача М. База знаний Холмса

Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

База знаний Холмса может быть смоделирована как набор выводов вида $A \rightarrow B$ (где A и B представляют события), что означает, что если произошло событие A , то событие B также должно было произойти. Назовем событие A возможной причиной события B . Конечно, выводы могут образовывать цепочки рассуждений, например $A \rightarrow B \rightarrow C$. Однако никогда не будет круговой цепи выводов (например $A \rightarrow B \rightarrow C \rightarrow \dots \rightarrow A$).

Холмсу дан набор $S = \{s_1, s_2, s_3, \dots, s_n\}$ событий, о которых известно, что они произошли. Затем он может, используя свои обширные знания и дедуктивные способности, найти все события, которые **точно** произошли.

Важно отметить, что знания Холмса настолько колоссальны, что он знает все возможные причины событий. Если событие произошло, и у события есть какие-то возможные причины, то хотя бы одна из этих причин тоже произошла. При этом бывают события, у которых нет причин, и такие события тоже могут происходить.

Многие детективные агентства высоко оценили бы уникальные способности Холмса, поэтому вам была поставлена задача выполнить с помощью компьютера то, что недоступно обычным смертным. Определите все события, которые точно произошли на основе базы знаний Холмса и собранных улик вашими коллегами-детективами.

Формат входных данных

Первая строка содержит три целых числа d, m, n ($1 \leq n \leq d \leq 1000$, $1 \leq m \leq 10^5$) — количество событий, выводов и собранных улик. Следующие m строк содержат по два целых числа A и B ($1 \leq A \neq B \leq d$), описывающих вывод $A \rightarrow B$. Последние n строк содержат по одному числу s_i ($1 \leq s_i \leq d$) — номера событий, про которые коллеги установили, что они точно произошли.

Формат выходных данных

Выведите номера событий, которые точно произошли, в порядке возрастания.

Примеры

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

Замечание

В первом примере из того, что произошло второе событие, можно сделать вывод, что третье тоже произошло. Причиной второго события могло быть только первое событие, так что оно тоже произошло. Во втором примере что-то послужило причиной для третьего события, но достоверно неизвестно, первое это событие или второе. В третьем примере, казалось бы, нельзя понять, что послужило причиной для четвертого события (второе или третье), но для каждого из них причиной было первое событие, значит, оно случилось, а значит, случились и второе с третьим.

Задача N. Массив с двойками

Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Загадан массив a , содержащий n чисел. У вас есть m наблюдений, каждое из которых состоит из трех элементов: двух индексов i, j , а также символа c . Если $i = j$, то символ c характеризует $v = a_i$, иначе c характеризует $v = a_i \cdot a_j$.

В зависимости от c про характеризующиеся значения v можно сделать следующие выводы:

$v = 0$		x
$0 < v < 1$		-
$v = 1$		=
$v > 1$		+

Ваша задача охарактеризовать каждое из значений массива a с помощью соответствующего $v = a_i$. Это возможно, ведь недавно стало известно, что все элементы массива a принадлежат множеству $\{0, \frac{1}{2}, 1, 2\}$.

Формат входных данных

Первая строка содержит два целых числа n и m ($1 \leq n \leq 10^5, 1 \leq m \leq 10^5$). Затем следуют m строк, каждая из которых содержит два целых числа i, j и символ c ($1 \leq i, j \leq n$ и c – один из x, -, = или +).

Формат выходных данных

Выведите одну строку с n символами, каждый из которых либо x, -, = или +. i -й символ должен описывать $v = a_i$ в формате, указанном выше.

Если существует несколько допустимых решений, вы можете вывести любое из них. Гарантируется, что существует хотя бы одно решение.

Пример

стандартный ввод	стандартный вывод
5 5 2 3 = 4 5 x 1 2 - 2 4 - 3 4 +	==+=x