

Задача А. Прямоугольник и квадраты

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

Илья зашел к своему другу Филу, и увидел у него удивительной красоты прямоугольник со сторонами a и b . Илья понял, что всю жизнь мечтал о прямоугольнике именно такой площади.

Придя домой, он увидел, что весь дом у него заполнен квадратами $C \times C$. Помогите ему собрать из квадратов $C \times C$ прямоугольник, как можно меньше отличающийся по площади от увиденного у Фила прямоугольника. А именно, он хочет, чтобы модуль разности площади прямоугольника Фила и собранного прямоугольника был как можно меньше.

Квадраты можно укладывать параллельно осям координат, без промежутков и наложений. Очевидно, что Илья возьмет хотя бы один квадрат.

Например, если у Фила прямоугольник имеет размер 4×5 , а у Ильи есть квадраты 3×3 , то прямоугольник с наиболее близкой площадью к прямоугольнику Фила, который может собрать Илья — 3×6 .

Формат входных данных

Входные данные содержат несколько тестовых наборов. В первой строке задано количество тестов t ($1 \leq t \leq 10000$).

Каждый тестовый пример описывается одной строкой, содержащей три целых числа A , B и C ($1 \leq A, B, C \leq 10^9$).

Формат выходных данных

Для каждого теста в отдельной строке выведите ответ на него — площадь прямоугольника, собранного Илеей из квадратов $C \times C$, как можно меньше отличающийся по площади от увиденного у Фила.

Если существует несколько ответов, выведите любой из них.

Пример

стандартный ввод	стандартный вывод
3	18
4 5 3	6
2 3 1	25
6 4 5	

Задача В. Нули и единицы

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

Сегодня Петя не сделал домашнее задание по информатике, поэтому его в качестве наказания вызвали к доске. Учитель написал на доске две строки одинаковой длины, состоящие из 0 и 1, и попросил сделать их равными, используя только одну операцию: разрешается взять два соседних символа любой из строк и инвертировать их, при инвертировании символ '0' становится равным '1', а '1' — '0'.

Также, чтобы усложнить задание и еще больше наказать Петю, учитель попросил сделать это за минимальное количество применений данной операции.

Петя в замешательстве и просит у вас помощи, помогите ему.

Например, пусть учитель написал на доске строки «0101» и «1111». Тогда один из оптимальных способов сделать их равными следующий: сначала инвертируем два центральных символа первой строки, получим строки «0011» и «1111». Теперь инвертируем два первых символа второй строки, получим строки «0011» и «0011». Заметим, что есть и другие способы добиться равных строк за два действия.

Формат входных данных

Входные данные содержат несколько тестовых наборов. В первой строке задано количество тестов t ($1 \leq t \leq 100$).

Каждый из следующих t тестов описывается следующим образом: в первой строке описания теста содержится число n ($1 \leq n \leq 10^5$) — длина строк, написанных учителем. Во второй и третьей строках описания теста содержатся строки, написанные учителем — две строки длины n из 0 и 1.

Гарантируется, что сумма чисел n по всем тестам не превосходит 10^5 .

Формат выходных данных

Для каждого теста в отдельной строке выведите ответ на него — минимальное количество применений операций, за которое можно сделать строки равными, или -1 , если сделать это невозможно.

Пример

стандартный ввод	стандартный вывод
2	2
4	-1
0101	
1111	
3	
011	
111	

Задача С. Новая трасса

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

Проектировщик трасс Гер Кильке получил заказ на новый трек Формулы-У. Гер планирует использовать свой любимый способ построения трасс. Он ввел на плоскости, где будет расположена трасса, систему координат, в которой ось X направлена слева направо, а ось Y направлена снизу вверх, и хочет построить трассу со следующими ограничениями:

- Трасса должна быть незамкнутой ломаной из n отрезков, параллельных осям координат.
- Один из концов ломаной является стартом, а другой — финишем, они должны находиться в разных точках.
- Концы отрезков ломаной должны находиться в точках с целыми положительными координатами, не превышающими 3000.
- Длины всех отрезков должны быть ненулевыми.
- При перемещении вдоль ломанной от старта к финишу угол между направлениями двух подряд идущих отрезков должен составлять 90 градусов по часовой стрелке.
- Конец любого отрезка не может находиться на другом отрезке, помимо предыдущего и следующего в ломаной. В частности, отрезки, которые не являются соседними в ломаной, не могут иметь общих фрагментов ненулевой длины.

Гер собирается составить новую трассу из n отрезков, так чтобы у неё было ровно k самопересечений. Математики объяснили ему, что это возможно, если $0 \leq k \leq n_2(n_2 - 1)/2$, где n_2 равно $n/2$, округленному вниз. Разумеется, Гер выбрал именно такое значение k . Помогите Геру спроектировать новую трассу.

Формат входных данных

Входные данные содержат несколько тестовых примеров. Первая строка файла содержит одно натуральное число t — число тестов.

Каждая из t следующих строк содержит два натуральных числа: n и k ($1 \leq n \leq 1000$, $0 \leq k \leq n_2(n_2 - 1)/2$, где n_2 равно $n/2$, округленному вниз) — число отрезков в ломаной и число самопересечений.

Сумма n по всем тестам одних входных данных не превосходит 10^4 .

Формат выходных данных

Выходной файл должен содержать ответы для каждого из t тестов.

Каждый ответ должен содержать $n+1$ строку, i -я из которых содержит координаты i -й вершины в ломаной. Выводить вершины следует в порядке обхода ломаной от старта к финишу. Координаты должны быть положительными целыми числами, не превышающим 3000.

Гарантируется, что при приведенных ограничениях искомая ломаная существует.

Пример

стандартный ввод	стандартный вывод
2	1 3
4 1	3 3
3 0	3 1
	2 1
	2 5
	3 9
	3 1
	1 1
	1 4

Задача D. Дерево

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

На День Рождения Коле подарили подвешенное дерево из n вершин.

Напомним, что подвешенное дерево — это связный неориентированный граф без циклов, в котором одна из вершин является корнем. Родителем вершины называется ее сосед, ближайший к корню, у корня нет родителя. Остальные соседи вершины называются ее детьми. Вершины подаренного Коле дерева пронумерованы от 1 до n , вершина с номером 1 является его корнем.

Коля решил поиграть со своим деревом. Он выбирает некоторую вершину u , удаляет ее и всех ее детей подвешивает к ее бывшему родителю u . Коля никогда не удаляет корень. Но просто так удалять вершины неинтересно, поэтому он попросил вас иногда находить расстояния между вершинами в текущем графе. Ответьте на все вопросы коварного Коли.

Формат входных данных

В первой строке находится одно целое число n ($1 \leq n \leq 100\,000$) — количество вершин в исходном дереве. Во второй строке находятся $n - 1$ целое число p_i ($1 \leq p_i \leq n$) — номера вершин, которые изначально являются родителями вершин $2, 3, \dots, n$.

В третьей строке находится одно целое число q ($1 \leq q \leq 100\,000$) — количество запросов. В следующих q строках находится описание запросов. Каждый запрос начинается с целого числа, которое равно 1 или 2. Если первое число в запросе равно 1, тогда далее следуют два числа a и b ($1 \leq a, b \leq n$) — две вершины, расстояние между которыми нужно найти. Иначе, далее следует одно число v ($1 \leq v \leq n$) — номер вершины, которую нужно удалить.

Гарантируется, что исходное дерево задано корректно, вершины в запросах ранее не были удалены, и корень никогда не удаляется.

Формат выходных данных

На каждый запрос второго типа в новой строке выведите расстояние между вершинами.

Пример

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

Задача Е. Варвары

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

Оказывается, что жизнь на островах иногда бывает не такой уж приятной! Недавно на страну, территория которой состоит из n островов, напали варвары. Острова этой страны были соединены мостами таким образом, что между двумя любыми островами существовал единственный путь, который не проходит через какой-либо остров дважды.

Варвары понимали, что мосты — слабое место этого государства. Поэтому они начали разрушать их один за другим! Недовольство жителей при этом росло и росло. Изначально, недовольство жителей острова i было равно a_i .

Рассмотрим, как меняется недовольство жителей некоторого острова x после разрушения очередного моста. Обозначим количество островов, до которых могли добраться жители x до разрушения моста как a , а после разрушения — b . Тогда недовольство возросло в $a - b + 1$ раз.

Вам предлагается разработать систему для определения текущего недовольства жителей. Вашей программе будут поступать запросы в формате пары чисел u и v . Такой запрос обозначает, что мост, который соединяет города $u + res$ и $v + res$ (где res — ответ на последний запрос), был разрушен. При этом надо считать, что вначале ответ на последний запрос равен 0.

Пусть после разрушения очередного моста недовольство жителей острова i равно b_i , тогда нужно сделать res равным остатку от деления суммы всех b_i на $10^9 + 7$, а также вывести это значение.

Для лучшего понимания формата входных и выходных данных рассмотрим, что происходит в примере.

Изначально недовольства жителей равны

1 2 3 4 5

После удаления моста между островами 1 и 3, недовольство жителей островов 1 и 2 возросло в 4 раза, а жителей других городов в 3 раза. Т. е. недовольства после разрушение моста стали равны

4 8 9 12 15

А суммарное недовольство всех равно 48. После этого был разрушен мост между $(-47) + 48 = 1$ и $(-46) + 48 = 2$. Соответственно, недовольства стали равны

8 16 9 12 15

А суммарное недовольство равно 60. После этого был разрушен мост, соединяющий острова 3 и 5. Недовольства стали равны

8 16 18 24 45

А суммарное недовольство равно 111. После разрушения последнего моста недовольства стали равны

8 16 36 48 45

Что в сумме равно 153.

Формат входных данных

В первой строке дано целое число n ($2 \leq n \leq 2 \cdot 10^5$) — количество островов.

В следующей строке задано n чисел a_i ($1 \leq a_i \leq 10^9 + 6$) — начальное недовольство жителей.

В следующих $n - 1$ строках заданы пары чисел u_i, v_i ($1 \leq u_i, v_i \leq n$). Такая пара обозначает, что острова u_i и v_i соединены мостом. Гарантируется, что от каждого острова до каждого другого можно дойти по мостам ровно одним способом.

В каждой из следующих $n - 1$ строк даны запросы в формате, который описан в условии. Для лучшего понимания смотрите объяснение к примеру в условии. Гарантируется, что любой мост существовал на момент его разрушения.

Формат выходных данных

Выведите ответ на каждый запрос в отдельной строке.

Пример

стандартный ввод	стандартный вывод
5	48
1 2 3 4 5	60
1 2	111
1 3	153
3 4	
3 5	
3 1	
-47 -46	
-57 -55	
-108 -107	