

Задача А. Целые точки

Имя входного файла: стандартный ввод или `input.txt`
Имя выходного файла: стандартный вывод или `output.txt`
Ограничение по времени: 2 секунды
Ограничение по памяти: 512 мегабайт

DLS и JLS скучают на уроке математики в школе. Чтобы как-то развлечь себя, DLS достал лист клетчатой бумаги и нарисовал на ней n различных прямых, заданных уравнениями $y = x + p_i$, для каждого i до 1 до n . JLS не заставил себя ждать и нарисовал на листе DLS m различных прямых, заданных уравнениями $y = -x + q_i$, для каждого i до 1 до m . DLS и JLS интересно, сколько пар прямых пересекаются в точках с целочисленными координатами. К сожалению, до конца урока осталось мало времени, поэтому DLS и JLS просят вас помочь им.

Формат входных данных

В первой строке задано целое число n ($1 \leq n \leq 100\,000$) — количество прямых, которые нарисовал DLS.

Во второй строке содержатся n различных целых чисел p_i ($0 \leq p_i \leq 10^9$), описывающих прямые, нарисованные DLS. Каждое из чисел описывает прямую, заданную уравнением $y = x + p_i$.

В третьей строке задано целое число m ($1 \leq m \leq 100\,000$) — количество прямых, которые нарисовал JLS.

В четвёртой строке содержатся m различных целых чисел q_i ($0 \leq q_i \leq 10^9$), описывающих прямые, нарисованные JLS. Каждое из чисел описывает прямую, заданную уравнением $y = -x + q_i$.

Формат выходных данных

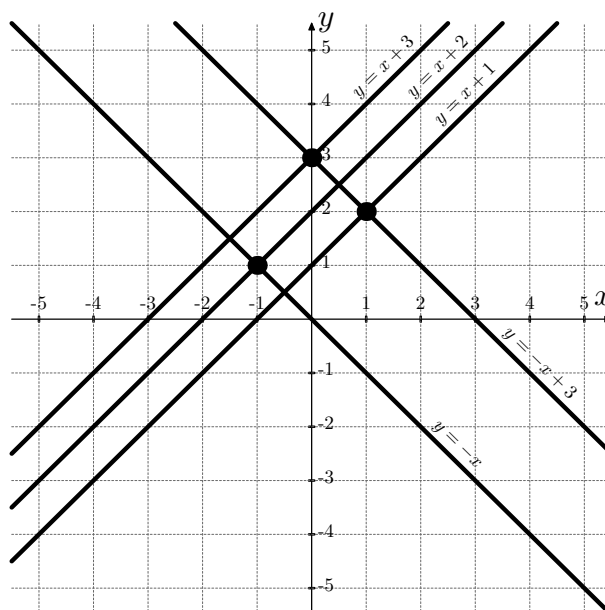
Выведите одно целое число — количество пар прямых, которые пересекаются в точке с целочисленными координатами.

Пример

стандартный ввод	стандартный вывод
3 1 3 2 2 0 3	3

Замечание

Ниже изображены прямые из первого примера. Чёрными кружочками выделены целочисленные точки пересечения.



Задача В. Фокус с делением и умножением

Имя входного файла:	стандартный ввод или <code>input.txt</code>
Имя выходного файла:	стандартный вывод или <code>output.txt</code>
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мегабайт

Наверняка вы знаете много математических фокусов. В этой задаче вам предлагается продемонстрировать свои умения в одном из таких фокусов.

У вас есть массив из различных натуральных чисел a_1, a_2, \dots, a_n . Помимо этого, перед вами стоит стол, на котором лежат 10^{18} карточек, на которых записаны все натуральные числа от 1 до 10^{18} , каждое ровно на одной карточке.

Вы можете подойти к столу и взять карточку с числом x . После этого вы можете выбрать ровно одно любое число a_i в вашем массиве и поменять его либо на $\frac{a_i}{x}$, либо на $a_i \cdot x$. При этом делить a_i на x разрешается только в случае, если a_i делится на x . Таким образом все числа в вашем массиве должны оставаться натуральными. После выполнения данной операции использованная карточка выкидывается и больше никогда не возвращается на стол.

Вы хотите несколько раз взять со стола карточки и выполнить операции таким образом, чтобы в итоге все числа массива оказались равными. Определите, какое минимальное количество карточек потребуется взять со стола, чтобы совершить этот фокус. Гарантируется, что хотя бы один способ сделать все числа в массиве равными существует.

Формат входных данных

В первой строке находится одно целое число n ($1 \leq n \leq 200\,000$) — количество чисел в вашем массиве.

Во второй строке находятся n целых чисел a_1, a_2, \dots, a_n ($1 \leq a_1 < a_2 < \dots < a_n \leq 10^{18}$), разделенных пробелами. Для удобства элементы массива даны в возрастающем порядке.

Формат выходных данных

Выведите минимальное количество карточек, которое нужно взять со стола, чтобы сделать все числа массива равными с помощью операций деления и умножения, описанных в условии задачи.

Примеры

стандартный ввод	стандартный вывод
2 1 2	1
5 2 3 6 12 18	5
3 239 717 1434	2

Замечание

В первом тесте, можно подойти к столу и взять карточку, на которой написано число 2. После этого можно выбрать элемент массива $a_2 = 2$ и заменить его на $\frac{2}{2} = 1$. Таким образом, все числа массива станут равны 1.

Во втором тесте можно подойти к столу пять раз и брать карточки, на которых записаны числа 32, 24, 12, 6, 4 и умножать на них первый, второй, третий, четвертый и пятый элемент массива, соответственно. Тогда все числа массива станут равны 72.

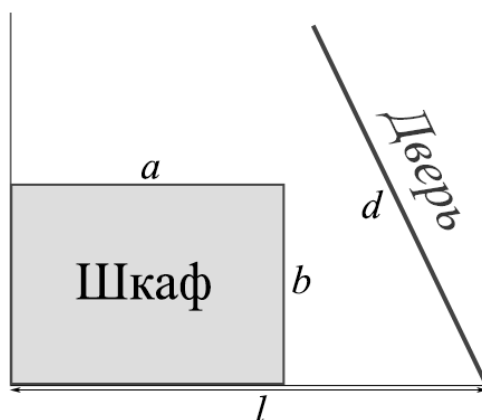
В третьем тесте можно подойти к столу, взять карточку с числом 3 и умножить a_1 на него. После этого можно взять карточку с числом 2 и поделить a_3 на него. Таким образом получится, что все числа массива равны 717.

Задача С. Дорогой шкаф

Имя входного файла:	стандартный ввод или <code>input.txt</code>
Имя выходного файла:	стандартный вывод или <code>output.txt</code>
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мегабайт

Владимир — успешный молодой человек, который всего за 3 года смог накопить достаточно денег и купить себе квартиру. Более того, у него осталась приличная сумма, чтобы купить себе в квартиру дорогой зеркальный шкаф. После долгих хождений по магазинам и поискам приличной мебели, Владимир наконец нашёл подходящий ему шкаф, имеющий форму прямоугольного параллелепипеда с длиной основания a и шириной основания b .

Владимир хочет поставить шкаф в угол своей комнаты прямо за дверью. Но вот незадача, дверь в квартире еще не установлена. Теперь Владимиру интересно, сможет ли дверь при открытии не задеть шкаф и упереться в стену, ведь в противном случае дорогое зеркало на шкафу может поцарапаться. Зная длину двери d , а также расстояние от двери до угла l , помогите Владимиру это выяснить. Даже если дверь немного коснётся шкафа, она его поцарапает, и в этом случае считается, что дверь задела шкаф. Дверь может упереться только в шкаф или в одну из двух стен, стыкующихся в углу шкафа под прямым углом. Две другие стены и вся остальная мебель находятся достаточно далеко. Для лучшего понимания ниже показан вид сверху на угол комнаты.



Формат входных данных

В четырёх строках заданы четыре целых числа a , b , d и l ($1 \leq a, b, d, l \leq 30\,000$, $a \leq l$) — длина шкафа, ширина шкафа, длина двери и расстояние от двери до угла комнаты, соответственно.

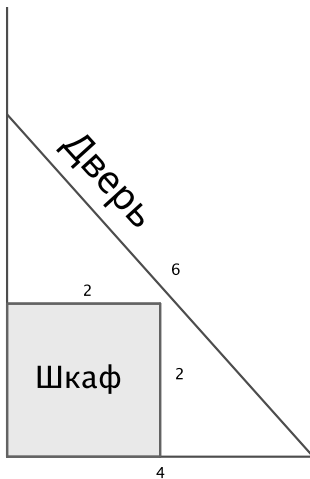
Формат выходных данных

В случае, если дверь упрётся в стену и не заденет шкаф, выведите «Yes». В противном случае выведите «No».

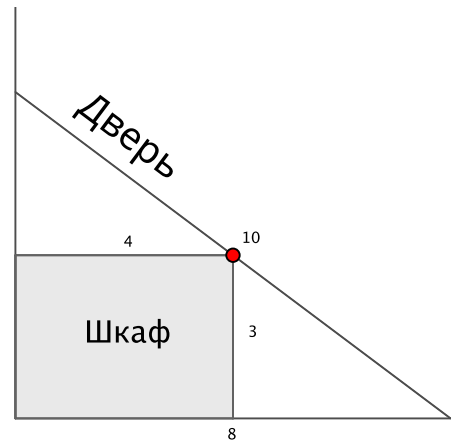
Примеры

стандартный ввод	стандартный вывод
2 2 6 4	Yes
4 3 10 8	No
1 1 1 2	No
1 1 1 3	Yes

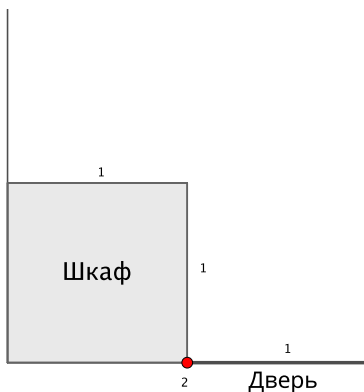
Замечание



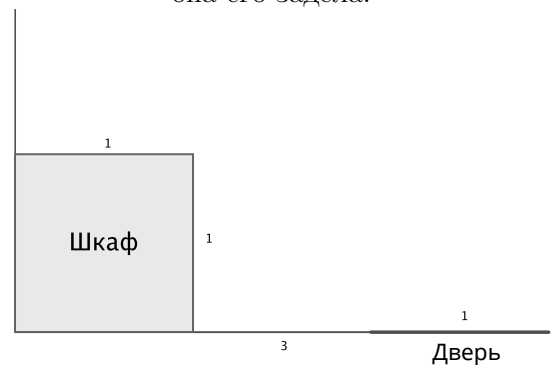
В первом примере дверь упрётся в дальнюю от неё стену и не заденет шкаф.



Во втором примере дверь упрётся в стену, при этом она коснётся шкафа, поэтому считается, что она его задела.



В третьем примере дверь упрётся в ближнюю стену, при этом её конец будет касаться шкафа, поэтому считается, что она его задела.



В четвёртом примере дверь упрётся в ближнюю стену, при этом её конец не будет касаться шкафа, поэтому считается, что она его не задела.

Задача D. Весь мир — задача по программированию

Имя входного файла:	стандартный ввод или <code>input.txt</code>
Имя выходного файла:	стандартный вывод или <code>output.txt</code>
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мегабайт

Вася — опытный составитель задач для олимпиад по программированию. На его счету множество разнообразнейших задач, с некоторыми из которых вы можете быть знакомы. Как и все великие творцы, Вася столкнулся с творческим кризисом. Уже который день Вася не выходит из комнаты и падает в пучину отчаяния и ужаса.

Чтобы исправить ситуацию, его друг Петя подарил ему строку, состоящую только из открывающих и закрывающих скобок. Петя рассказал, что красотой строки из скобок называется количество её циклических сдвигов, которые приводят к правильной скобочной последовательности.

Напомним, что правильной скобочной последовательностью называется скобочная последовательность, которая может быть получена из корректного арифметического выражения, содержащего круглые скобки, путем удаления всех символов, кроме скобок. Например, скобочные последовательности «`()()`», «`((()))`» — правильные (примеры исходных выражений: «`(1+1)+(1+1)`», «`((1+1)+1)`»), а «`)()`» и «`(`» — нет.

Циклическим сдвигом строки s длины n на k ($0 \leq k < n$) называется строка, представляющая собой конкатенацию (сложение) последних k символов строки (суффикса длины k) s и первых $n - k$ символов строки (префикса длины $n - k$). Два циклических сдвига на i и j называются различными, если $i \neq j$.

Чтобы отвлечься от проблем, Вася хочет взять любые два символа (**не обязательно различных**) и поменять их местами. Ему стало интересно, какой максимальной красоты строки можно добиться, применив данную операцию. Помогите ему.

Формат входных данных

В первой строке задано одно целое число n ($1 \leq n \leq 300\,000$) — длина строки, которую подарили Васе.

Во второй строке задана строка, состоящая ровно из n символов, каждый из которых — либо открывающая скобка «`(`», либо закрывающая скобка «`)`».

Формат выходных данных

В первой строке выведите одно число — максимальную красоту строки, которую можно получить, поменяв местами какие то два символа. Во второй строке выведите два числа l и r ($1 \leq l, r \leq n$) — номера символов, которые нужно поменять местами, чтобы максимизировать красоту строки. Если ответов несколько, разрешается вывести любой.

Примеры

стандартный ввод	стандартный вывод
10 <code>()()()()</code>	5 8 7
12 <code>)()()()()</code>	4 5 10
6 <code>)))()</code>	0 1 1

Замечание

В первом примере после обмена местами 7-й и 8-й скобок получается строка «`()()()()()`», её циклические сдвиги на 0, 2, 4, 6, 8 являются правильными скобочными последовательностями.

Во втором примере после обмена местами 5-й и 10-й скобок получается строка «`)()()()()`», её циклические сдвиги на 11, 7, 5, 3 являются правильными скобочными последовательностями.

В третьем примере какие бы две скобки мы не поменяли местами, число циклических сдвигов, являющихся правильными скобочными последовательностями, будет равно 0.

Задача Е. Черепашка

Имя входного файла: стандартный ввод или `input.txt`
Имя выходного файла: стандартный вывод или `output.txt`
Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

У мальчика Коля есть черепашка и поле $2 \times n$. Строки поля пронумерованы от 1 до 2 сверху вниз, а столбцы от 1 до n слева направо.

Предположим, что в каждой клетке есть съедобный лист салата, энергетическая ценность которого в клетке поля в строке i и столбце j составляет $a_{i,j}$ ($a_{i,j}$ равно нулю или единице). Черепашка изначально находится в верхней левой клетке и хочет попасть в правую нижнюю. Черепашка умеет двигаться только вниз и вправо, а среди всех возможных путей она всегда выберет тот, который максимизирует суммарную энергетическую ценность листьев салата, через которые она проходит (если таких путей несколько, она выберет произвольный из них).

Коля беспокоится, что если черепашка съест слишком много салата, то это может быть опасно для её здоровья. Поэтому он хочет переставить листья салата таким образом, чтобы минимизировать суммарную энергетическую ценность салата, которую съест черепашка. Обратите внимание, что минимизировать количество перестановок листьев салата не требуется, то есть Коля просто соберёт все листья со всех клеток и разложит их заново, так чтобы в каждой клетке оказался ровно один лист.

Формат входных данных

Первая строка содержит одно целое число n ($2 \leq n \leq 100\,000$) — длину поля.

Вторая строка содержит n целых чисел $a_{1,i}$ ($0 \leq a_{1,i} \leq 1$), описывающих энергетическую ценность листов салата в первой строке поля.

Третья строка содержит n целых чисел $a_{2,i}$ ($0 \leq a_{2,i} \leq 1$), описывающих энергетическую ценность листьев салата во второй строке поля.

Формат выходных данных

Выведите две строки по n чисел в каждой — расстановку листьев салата после переупорядочивания Коля.

Если существует несколько оптимальных переупорядочиваний — выведите любое из них.

Примеры

стандартный ввод	стандартный вывод
3 0 0 0 0 0 0	0 0 0 0 0 0
3 1 0 1 0 0 0	0 0 1 0 1 0

Замечание

В первом примере черепашка съест салат суммарной энергетической ценности 0.

Во втором примере черепашка сможет съесть салат суммарной энергетической ценности 1.

Задача F. Иванушка-дурачок и теория вероятностей

Имя входного файла:	стандартный ввод или <code>input.txt</code>
Имя выходного файла:	стандартный вывод или <code>output.txt</code>
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мегабайт

Недавно Иванушка-дурачок захотел стать умнее и поэтому начал изучать теорию вероятностей. Он считает, что уже достаточно хорошо понимает данную тему, и стал этим хвастаться перед всеми.

Чтобы не быть голословным, Иванушка пообещал продемонстрировать окружающим свои навыки при помощи случайного рисунка. Рисунок представляет собой клетчатое поле с n строками и m столбцами, каждая клетка которого покрашена в белый или чёрный цвет. Иванушка считает, что рисунок случайный, если для любой клетки верно, что среди ее соседей по стороне не более одной клетки такого же цвета, что и она.

Его братья быстро выяснили, что задумал Иванушка. Они долго пытались объяснить ему, что он, дурачок, все ещё плохо понимает, что такое случайность. Но Иванушка так и не поверил им, поэтому они попросили помощи у вас. Они хотят найти количество различных случайных по мнению Иванушки рисунков. Два рисунка называются различными если существует клетка, которая покрашена в разные цвета на этих рисунках. Так как это число может быть слишком большим, найдите остаток от деления его на $10^9 + 7$.

Формат входных данных

В единственной строке заданы два целых числа n и m ($1 \leq n, m \leq 100\,000$) — количество строк и столбцов клетчатого поля, соответственно.

Формат выходных данных

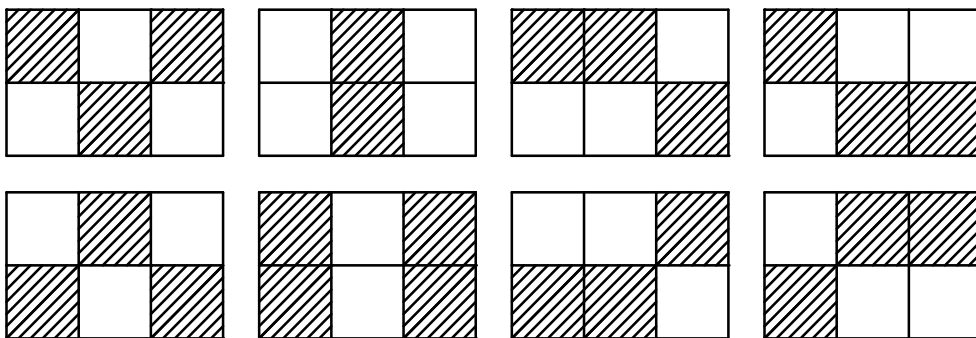
Выведите одно целое число — остаток от деления числа случайных рисунков на $10^9 + 7$.

Пример

стандартный ввод	стандартный вывод
2 3	8

Замечание

На картинке ниже изображены все возможные случайные раскраски поля 2 на 3.



Задача Г. Выращивание дерева

Имя входного файла: стандартный ввод или `input.txt`
Имя выходного файла: стандартный вывод или `output.txt`
Ограничение по времени: 2 секунды
Ограничение по памяти: 512 мегабайт

Садоводу Лёше, преподающему школьникам олимпиадную информатику, на день учителя подарили набор деревянных палочек, каждая из которых представляет собой отрезок целочисленной длины. Теперь Лёша хочет вырастить из них дерево.

Дерево представляет собой ломаную на плоскости, состоящую из подаренных ему палочек. Ломаная начинается в точке $(0, 0)$. Строя ломаную, Лёша может соединять палочки в любом порядке, располагая каждую горизонтально или вертикально (то есть параллельно осям OX и OY). При этом он не может располагать две идущие подряд палочки одновременно горизонтально или вертикально. Лёша хочет сделать так, чтобы конец ломаной был как можно дальше от точки $(0, 0)$. Помогите ему вырастить дерево таким образом!

Формат входных данных

Первая строка содержит целое число n ($1 \leq n \leq 100\,000$) — количество палочек, подаренных Лёше.

Вторая строка содержит n целых чисел a_1, \dots, a_n ($1 \leq a_i \leq 10\,000$) — длины палочек, подаренных Лёше.

Формат выходных данных

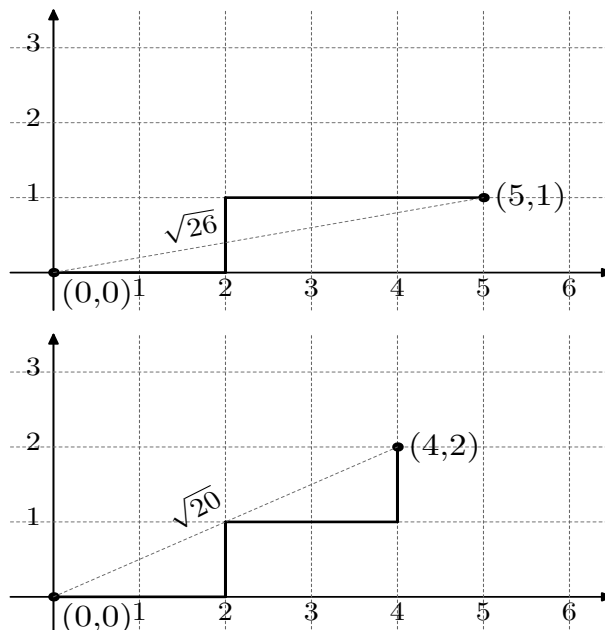
В единственной строке выведите одно целое число — **квадрат** максимального расстояния от точки $(0, 0)$ до конечной точки дерева.

Примеры

стандартный ввод	стандартный вывод
3 1 2 3	26
4 1 1 2 2	20

Замечание

На следующих изображениях нарисованы оптимальные деревья для примеров из условия. Квадрат расстояния в первом случае равен $5 \cdot 5 + 1 \cdot 1 = 26$, а во втором — $4 \cdot 4 + 2 \cdot 2 = 20$.



Задача Н. Дороги в стране

Имя входного файла:	стандартный ввод или <code>input.txt</code>
Имя выходного файла:	стандартный вывод или <code>output.txt</code>
Ограничение по времени:	2 секунды
Ограничение по памяти:	512 мегабайт

В некоторой стране есть n городов, пронумерованных целыми числами от 1 до n . Столица этой страны расположена в городе с номером 1. Некоторые пары городов в этой стране соединены дорогами с односторонним движением, при этом для каждой пары городов u и v есть не более одной дороги, ведущей из города u в город v , однако наличие дороги из города u в город v не запрещает существование дороги из города v в город u . Для каждой дороги известно, груз какого максимального веса можно по ней перевозить. Требуется для каждого города определить максимальный вес груза, который можно доставить из него в столицу.

Формат входных данных

В первой строке содержатся два целых числа n и m ($2 \leq n \leq 10^5$, $0 \leq m \leq 10^6$) — количество городов и дорог в стране, соответственно.

В следующих m строках содержатся описания дорог. Описание дороги состоит из трёх целых чисел u , v и w ($1 \leq u, v \leq n$, $u \neq v$, $1 \leq w \leq 10^9$), описывающих дорогу с односторонним движением из города с номером u в город с номером v , по которой можно перевозить грузы весом не более w . Гарантируется, что для каждой пары городов u и v существует не более одной дороги из u в v .

Формат выходных данных

Выведите $n - 1$ целых чисел. i -е число должно быть равно максимальному весу груза, который можно перевезти из города $i + 1$ в столицу (город с номером 1). Если из города $i + 1$ в столицу доставить груз невозможно, то выведите -1 .

Примеры

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

Замечание

В первом примере есть только одна дорога, заканчивающаяся в столице, и по ней нельзя везти грузы весом более 1, при этом из всех городов можно доехать до столицы по дорогам. Таким образом, из всех городов можно привезти в столицу груз весом не более 1.

Во втором примере из города 2 в город 1 можно перевезти груз весом 3 следующим образом: проехать из города 2 в город 3 по дороге, по которой можно везти грузы весом не более 3, а затем из города 3 в столицу (город 1) по дороге, по которой можно везти грузы весом не более 7. Из города 3 в город 1 можно доставить груз весом 7 по прямой дороге, соединяющей эти города. Аналогично, из города 4 в город 1 можно перевезти груз весом 1 по дороге, соединяющей эти города напрямую.

В третьем примере из города 3 в столицу доехать нельзя.