

Задача А. Польшар и Подарки

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

Все задачи этого конкурса по уровню как В, С или D региона, ничего сложнее тут нет. Если вы хотите подготовиться к региону, советуем прорешать все задачи.

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 подарок.

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 В. Украденный массив

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

Мальчик Петя любит массивы. Недавно ему подарили огромный массив чисел F размера 2^n . Петя человек странный, и при виде массива из чисел сразу начинает считать какие-то суммы на нём. Специально для этого он купил в магазине новый пустой массив P размера 2^n и начал его заполнять по следующему правилу: $P[i] = \sum_{j \& i = j} F[j]$. Другими словами для каждого j , такого что j — подмаска i (т.е. побитовое «И» чисел i и j равно j), Петя прибавил $F[j]$ к изначально нулевому значению $P[i]$.

Но потом случилось ужасное — массив F украли! Теперь Петя хочет разыскать массив F , но он не помнит, какие значения там были изначально. Единственное что у него есть — массив P . Помогите Пете восстановить массив F .

Формат входных данных

В первой строке входных данных дано одно число n ($1 \leq n \leq 20$).

В следующей строке даны 2^n чисел, i -е из них — значение $P[i]$ ($-10^9 \leq P[i] \leq 10^9$) (нумерация ведётся с нуля).

Формат выходных данных

В одной строке выведите 2^n чисел, i -е из которых — значение $F[i]$ (нумерация ведётся с нуля).

Примеры

стандартный ввод	стандартный вывод
2 1 2 3 4	1 1 2 0
3 1 3 4 10 6 14 16 36	1 2 3 4 5 6 7 8

Задача С. Путьевые строки

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

Рассмотрим ориентированный граф G , имеющий n вершин, пронумерованных натуральными числами от 1 до n . В графе G возможно наличие нескольких дуг между одной и той же парой вершин, а также дуг, ведущих из вершины в нее саму. Каждая дуга графа помечена некоторой буквой латинского алфавита. Каждому пути в графе G можно поставить в соответствии строку, состоящую из букв, написанных на последовательно проходимых в соответствии с этим путем дугах. Эта строка называется путевой меткой пути. Назовем строку S путевой строкой графа G , если в нем существует путь, путевая метка которого равна S .

Ваша задача посчитать остаток от деления на 1 000 000 количества путевых строк графа G , состоящих ровно из L символов.

Формат входных данных

В первой строке входных данных записаны целые числа n, m, L ($1 \leq n \leq 10$, $1 \leq m \leq 10\,000$, $1 \leq L \leq 100$), равные количеству вершин и ребер графа G , а также длине путевых строк, которые нужно искать. Следующие m строк задают дуги графа G . Каждая из этих строк содержит два натуральных числа a, b ($1 \leq a, b \leq n$) и маленькую латинскую букву c , что означает наличие дуги из вершины a в вершину b , помеченной символом c . Элементы каждой строки отделены друг от друга пробелами.

Формат выходных данных

Единственная строка выходных данных должна содержать одно число, равное остатку от деления количества путевых строк длины L в графе G на 1 000 000.

Пример

стандартный ввод	стандартный вывод
4 4 100 1 2 a 2 3 b 3 4 a 4 1 b	2

Задача D. Коровники

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

В селе Максоярославке коровы обычно пасутся на лужайках, соединенных дорожками, на каждой лужайке пасется хотя бы одна корова. При этом для каждой пары лужаек есть ровно один способ пройти от одной лужайки до другой. По каждой дорожке можно двигаться в обоих направлениях. Считается, что все дорожки имеют одинаковую длину.

Главный фермер села хочет построить на лужайках k коровников для своих коров. Ясно, что каждая корова вечером будет возвращаться именно в тот коровник, который ближе к ее лужайке (если расстояние до коровников одинаково, то в любой из них). Поэтому возникает задача определения такого расположения коровников, при котором наибольшее из расстояний, проходимых коровами, было бы минимально.

Формат входных данных

В первой строке входного файла содержатся два числа n и k ($2 \leq n \leq 50\,000$, $1 \leq k \leq n$) — количество лужаек и планируемое число коровников. Следующие $n - 1$ строк содержат описания дорожек. Каждая дорожка задается парой целых положительных чисел (a, b) , где a и b — номера лужаек, которые соединяет данная дорожка. Лужайки нумеруются с единицы.

Формат выходных данных

В первой строке входного файла выведите l — максимальное количество дорожек, по которым придется пройти корове, чтобы попасть в коровник.

Пример

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

Задача Е. Поздний контекст

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

Много в мире разных часовых поясов! Именно поэтому соревнования по программированию часто бывают в неудобное для некоторых людей время. Как-то раз Гриша и Егор решили поучаствовать в соревновании, которое заканчивалось очень поздно. Именно поэтому ребята решили переночевать в гостинице, чтобы не возвращаться домой поздно. Однако все не так просто. Родители Егора и Гриши очень волнуются за своих детей, поэтому они решили установить по всему городу камеры, чтобы видеть, где находятся ребята.

В городе, где живут программисты $1 \leq n \leq 500\,000$ перекрестков, соединенных $n - 1$ дорогой так, что между любыми двумя перекрестками существует путь по дорогам.

Родители собираются установить на перекрестках города камеры, радиус действия которых равен длине дороги. Родители будут спокойны, если смогут видеть ребят на любом перекрестке. Иными словами, для каждого перекрестка должен существовать перекресток, находящийся на расстоянии не более одной дороги, такой что на нем установлена камера. Установка камер — затратное дело, поэтому для каждого перекрестка с номером i известна стоимость $1 \leq cost_i \leq 10^9$ установки камеры на нем.

Помогите родителям установить камеры надлежащим образом за минимальную стоимость.

Формат входных данных

В первой строке входного файла содержится количество перекрестков n ($1 \leq n \leq 500\,000$).

В последующих $n - 1$ строках содержатся $v_i u_i$ — перекрестки соединенные очередной дорогой.

Последняя строчка входного файла содержит n чисел $cost_1, cost_2, \dots, cost_n$ ($1 \leq cost_i \leq 10^9$) — стоимости установки камер на перекрестках.

Формат выходных данных

В первой строке выведете минимальную стоимость установки ans и количество перекрестков k , в которых надо установить камеры.

Во второй строке выведите k чисел — перекрестки, в которых надо установить камеры.

Если ответов несколько, разрешается вывести любой из них.

Примеры

стандартный ввод	стандартный вывод
6 1 2 2 3 1 4 4 5 4 6 228 1488 2 2 8 1	232 3 1 3 4
1 228	228 1 1

Задача F. Деревни

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

В тридесятом государстве есть N деревень. Некоторые пары деревень соединены дорогами. В целях экономии, «лишних» дорог нет, т.е. из любой деревни в любую можно добраться по дорогам единственным образом.

Новейшие исследования показали, что тридесятое государство находится в сейсмически опасной зоне. Поэтому глава государства захотел узнать, какой именно ущерб может принести его державе землетрясение. А именно, он хочет узнать, какое минимальное число дорог должно быть разрушено, чтобы образовалась изолированная от остальных группа ровно из P деревень такая, что из любой деревни из этой группы до любой другой деревни из этой группы по-прежнему можно будет добраться по неразрушенным дорогам (группа изолирована от остальных, если никакая неразрушенная дорога не соединяет деревню из этой группы с деревней не из этой группы).

Вы должны написать программу, помогающую ему в этом.

Формат входных данных

Первая строка входного файла содержит два числа: N и P ($1 \leq P \leq N \leq 150$). Все остальные строки содержат описания дорог, по одному на строке: описание дороги состоит из двух номеров деревень (от 1 до N), которые эта дорога соединяет. Все числа во входном файле разделены пробелами и/или переводами строки.

Формат выходных данных

В выходной файл выведите единственное число — искомое количество дорог.

Пример

стандартный ввод	стандартный вывод
11 6 1 2 1 3 1 4 1 5 2 6 2 7 2 8 4 9 4 10 4 11	2

Задача G. Реки

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

Почти все Королевство Байтленд покрыто лесами и реками. Малые реки сливаются в более крупные реки, которые, в свою очередь, сливаются друг с другом; в конечном счете, все реки сливаются вместе в одну большую реку. Большая река впадает в море вблизи города Байттаун.

В Байтленде имеется n лесозаготовительных поселков, каждый из которых расположен вблизи какой-либо реки. В настоящее время в Байттауне находится большая пилорама, которая обрабатывает все деревья, срубленные в Королевстве. Деревья сплавляются вниз по рекам от поселков, где они срублены, к пилораме в Байттауне. Король Байтленда решил поставить k дополнительных пиlorам в поселках, чтобы уменьшить стоимость сплава деревьев. После установки пиlorам деревья не обязательно должны сплавляться в Байттаун, а могут быть обработаны на ближайшей пиlorаме, находящейся ниже по течению рек. Очевидно, что деревья, срубленные в окрестности поселка с пиlorамой, вообще не сплавляются по рекам.

Необходимо отметить, что реки в Байтленде не разветвляются. Из этого следует, что для каждого поселка существует единственный путь сплава деревьев вниз по течению рек от него в Байттаун.

Королевские счетоводы подсчитали количество деревьев, срубаемых в каждом поселке за год. Вам необходимо определить, в каких поселках следует установить пиlorамы, чтобы минимизировать общую стоимость сплава деревьев за год. Стоимость сплава одного дерева составляет один цент за каждый километр пути.

Формат входных данных

Первая строка входных данных содержит два целых числа: n — количество поселков, не считая Байттауна ($2 \leq n \leq 100$), и k — количество дополнительных пиlorам, которые будут установлены ($1 \leq k \leq 50$ и $k \leq n$). Поселки нумеруются числами $1, 2, \dots, n$, а Байттаун имеет номер 0.

Каждая из последующих n строк содержит три целых числа, разделенных одним пробелом. Строка $i + 1$ содержит w_i — количество деревьев, срубаемых в поселке i за год ($0 \leq w_i \leq 10\,000$), v_i — ближайший поселок (либо Байттаун) вниз по реке от поселка i ($0 \leq v_i \leq n$), d_i — расстояние (в километрах) по реке от поселка i до поселка v_i ($1 \leq d_i \leq 10\,000$).

Гарантируется, что суммарная стоимость сплава всех деревьев к пиlorаме в Байттауне не превосходит 2 000 000 000 центов в год.

Формат выходных данных

Первая и единственная строка выходных данных должна содержать одно целое число: минимальную стоимость сплава (в центах).

Пример

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

Задача N. Два пути

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

Как вы знаете, Васин брат живет во Флатландии. Во Флатландии n городов, соединенных $n - 1$ двухсторонней дорогой. Города пронумерованы от 1 до n . Из любого города можно добраться до любого другого.

Компания «Два пути», в которой работает Васин брат, выиграла тендер на ремонт двух путей во Флатландии. Путем называется последовательность различных городов, последовательно соединенных дорогами. Пути, которые надо отремонтировать, компания может выбрать самостоятельно. Единственное условие, накладываемое тендером, они не должны пересекаться (то есть иметь общих городов).

Известно, что прибыль, которую получит компания «Два пути», равна произведению длин выбранных двух путей. Считая длину каждой дороги один, а длину пути равной количеству дорог в ней, найдите максимальную возможную прибыль.

Формат входных данных

В первой строке записано целое число n ($2 \leq n \leq 100\,000$), где n количество городов в стране. Далее в $n - 1$ строке записаны сами дороги. Каждая строка содержит пару номеров соединяемых дорогами a_i, b_i ($1 \leq a_i, b_i \leq n$)

Формат выходных данных

Выведите максимально возможную прибыль.

Примеры

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

Задача I. Мониторинг труб

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

Газораспределительная система одного региона устроена следующим образом. Она содержит n узлов, пронумерованных от 1 до n , некоторые узлы соединены односторонними трубами. Узел с номером 1 соответствует центральному газохранилищу.

Система узлов описывается числами от p_2, p_3, \dots, p_n . Для всех i от 2 до n узел с номером p_i соединен односторонней трубой с узлом i , газ по этой трубе передается от узла p_i к узлу i . Известно, что возможно доставить газ по трубам от центрального газохранилища до любого узла системы (возможно, с использованием промежуточных узлов). В системе используются трубы различных типов, тип трубы обозначается буквой английского алфавита от «a» до «z». Труба, соединяющая узел p_i с узлом i , имеет тип c_i .

Для проверки качества труб используется специальный робот. Он помещается в систему труб в одном из узлов и перемещается по трубам, каждый раз проверяя трубу, по которой он перемещается. Робот может перемещаться по трубам только в том же направлении, в котором по трубе передается газ. Совершив одно или несколько перемещений по трубам между узлами, робот извлекается из системы труб.

Каждый запуск робота должен соответствовать одной из m заданных спецификаций, пронумерованных от 1 до m . Спецификация с номером t представляет собой строку st , состоящую из строчных букв английского алфавита. Запуск соответствует спецификации st , если количество перемещений робота по трубам во время запуска совпадает с длиной st , и для всех j от 1 до длины st на j -м шаге робот перемещается по трубе, тип которой совпадает с $st[j]$ — символом на позиции j в спецификации.

Если запуск робота соответствует спецификации с номером t , то стоимость этого запуска составляет w_t . Оператору системы необходимо проверить все трубы, для этого можно запускать робот несколько раз. Каждый раз выбирается спецификация и маршрут робота по трубам, соответствующие выбранной спецификации. Необходимо проверить все трубы так, чтобы суммарная стоимость запусков робота для проверки качества труб была минимальна. Одну и ту же трубу можно проверить несколько раз.

Требуется написать программу, которая по описанию системы труб и списку спецификаций определяет минимальную суммарную стоимость запусков робота, в результате которых все трубы будут проверены, а также список необходимых для этого запусков (по требованию).

Формат входных данных

В первой строке входных данных находятся три целых числа n , m и t — количество узлов системы труб, количество спецификаций запусков робота и параметр, указывающий, требуется ли вывести список запусков робота или только их минимальную суммарную стоимость ($1 \leq n \leq 500$, $1 \leq m \leq 10^5$, t равно 0 или 1).

В последующих $(n - 1)$ строках содержится информация о трубах, $(i - 1)$ -я из этих строк содержит разделенные пробелом значения p_i и c_i , где p_i — целое число, задающее номер узла, из которого ведет труба в i -й узел, а c_i — строчная буква английского алфавита, задающая тип этой трубы ($1 \leq p_i \leq i - 1$).

В последующих m строках содержится информация о спецификациях, i -я из этих строк содержит разделенные пробелом целое число w_i — стоимость запуска робота в соответствии с этой спецификацией, и состоящую из строчных букв английского алфавита строку s_i — саму спецификацию ($1 \leq w_i \leq 10^9$). Суммарная длина строк s_i не превышает 10^6 .

Формат выходных данных

Первая строка выходных данных должна содержать одно число — минимальную суммарную стоимость запусков робота, в результате которых все трубы будут проверены. Если проверить все трубы невозможно, требуется вывести «-1».

Если $t = 0$, то больше ничего выводить не требуется.

Если $t = 1$ и проверить трубы возможно, то далее следует вывести список описаний запусков робота. В этом случае вторая строка выходных данных должна содержать число k — количество запусков робота, которое необходимо выполнить для проверки труб. В следующих k строках необходимо вывести по три целых числа a_i , b_i и c_i — номер узла, в котором начинается запуск, номер узла, в котором заканчивается запуск, и номер спецификации, которой соответствует запуск. Если оптимальных способов проверки несколько, требуется вывести любой из них.

Примеры

стандартный ввод	стандартный вывод
3 3 0 1 a 2 b 3 a 4 b 2 a	6
7 3 1 1 a 2 a 3 b 3 b 1 b 6 b 3 aab 5 b 2 ab	15 4 1 4 1 1 6 2 6 7 2 2 5 3