

Задача А. Центроиды дерева

Имя входного файла: centroid.in
Имя выходного файла: centroid.out
Ограничение по времени: 5 секунд
Ограничение по памяти: 256 мегабайт

Пожалуйста, напишите алгоритм с нуля, чтобы убедиться в своей компетентности.

Дано дерево из n вершин. У каждой вершины есть цвет. Нужно обработать q запросов (v_i, c_i) : найти расстояние от v_i до ближайшей к v_i вершины цвета c_i . Расстоянием между вершинами называется минимальное количество рёбер в пути между ними.

Формат входных данных

На первой строке число n ($1 \leq n \leq 10^5$), следующая строка содержит числа $p_1, p_2, \dots, p_{n-1}$. $0 \leq p_i < i$. p_i – отец вершины i в дереве. Далее строка с числами $a_0, a_1, \dots, a_{n-1}$. $0 \leq a_i < n$. a_i – цвет вершины i . Далее строка с числом q ($1 \leq q \leq 10^5$). Следующие q строк содержат запросы $v_i q_i$ ($0 \leq v_i < n$, $0 \leq c_i < n$).

Формат выходных данных

Для каждого запроса выведите одно число – расстояние до ближайшей вершины нужного цвета, или -1 , если в дереве нет вершин такого цвета.

Примеры

centroid.in	centroid.out
5	0 1 2 -1 2 1 2 1 1
0 1 1 3	
1 2 3 2 1	
9	
0 1	
0 2	
0 3	
1 0	
2 1	
2 2	
3 3	
3 1	
4 2	

Задача В. Почтовая реформа

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

Пожалуйста, напишите алгоритм с нуля, чтобы убедиться в своей компетентности.

В Флатландии идет пора реформ. Недавно была проведена реформа дорог, так что теперь по дорогам страны из любого города можно добраться в любой другой, причем только одним способом. Также была проведена реформа волшебников, так что в каждом городе остался ровно один волшебник. Теперь же началась реформа почтовой системы.

Недавно образованное почтовое агентство «Экс-Федя» предлагает уникальную услугу — коллективную посылку. Эта услуга позволяет отправлять посылки жителям всех городов на каком-либо пути по цене обычной посылки. Удивительно, но пользоваться такой услугой стали только волшебники Флатландии, которые стали в большом количестве отправлять друг другу магические кактусы. Агентство столкнулось с непредвиденной проблемой: как известно, все волшебники живут в башнях и мало того, что не строят в них лестницы, так еще время от времени меняют их высоту. Поэтому, чтобы доставить посылку волшебнику, который живет в башне высотой h , курьеру агентства требуется иметь с собой не менее h метров веревки.

Вам поручено руководить отделом логистики — по имеющимся данным о высотах башен и об их изменениях вам нужно определять минимальную длину веревки, которую нужно выдать курьеру, который доставляет посылки между городами i и j .

Формат входных данных

Первая строка входного файла содержит число n — количество городов в Флатландии ($1 \leq n \leq 50\,000$). Во второй строке находится n положительных чисел, не превосходящих 10^5 — высоты башен в городах. В следующих $n - 1$ строках содержится по два числа u_i и v_i — описание i -й дороги, $1 \leq u_i, v_i \leq n, u_i \neq v_i$. В следующей строке содержится число k — количество запросов ($1 \leq k \leq 100\,000$). В следующих k строках содержатся описания запросов в следующем формате:

- Уведомление от волшебника из города i о том, что высота его башни стала равна h , имеет вид $! i h$, $1 \leq i \leq n, 1 \leq h \leq 10^5$.
- Запрос от курьера о выдаче веревки для доставки посылок во все города на пути от i до j включительно имеет вид $? i j$, $1 \leq i, j \leq n$.

Формат выходных данных

Для каждого запроса доставки посылок выведите минимальную длину веревки, которую необходимо выдать курьеру.

Примеры

стандартный ввод	стандартный вывод
3 1 2 3 1 3 2 3 5 ? 1 2 ! 1 5 ? 2 3 ! 3 2 ? 1 2	3 3 5
1 100 5 ! 1 1 ? 1 1 ! 1 1000 ? 1 1 ! 1 1	1 1000

Задача С. Разбиение дерева

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

Дано дерево из n вершин, пронумерованных от 1 до n . Рёбра дерева пронумерованы от 0 до $n - 2$. Требуется удалить некоторое множество рёбер (возможно, пустое) так, чтобы каждая связная компонента полученного графа содержала количество вершин, кратное заданному числу t . Среди всех таких множеств удаляемых рёбер необходимо выбрать то, которое максимизирует число получившихся компонент. Необходимо вывести максимально возможное число компонент и само множество удаляемых рёбер.

Гарантируется, что решение всегда существует (то есть можно удалить ноль или более рёбер так, чтобы все компоненты имели размер, кратный t).

Формат входных данных

Первая строка содержит два целых числа n и t ($1 \leq t \leq n \leq 100\,000$).

В следующих $n - 1$ строках заданы рёбра. Для каждого i от 0 до $n - 2$ даны два целых числа a_i, b_i ($1 \leq a_i, b_i \leq n, a_i \neq b_i$) — номера вершин, соединяемых ребром с номером i .

Формат выходных данных

В первой строке выведите одно целое число — максимальное количество компонент.

Во второй строке выведите через пробел номера рёбер, которые необходимо удалить для достижения этого максимума в произвольном порядке. Рёбра нумеруются с 0.

Примеры

стандартный ввод	стандартный вывод
15 5 1 9 9 12 12 13 12 15 2 3 4 2 8 4 7 5 5 6 8 11 11 14 11 10 5 4 9 4	2 13
5 1 1 2 2 3 3 4 4 5	5 3 2 1 0

Замечание

В первом примере удаление ребра с номером 13 (соединяющего вершины 9 и 4) разбивает дерево на две компоненты: в одной 5 вершин, в другой 10 — обе кратны 5. Больше чем 2 компоненты получить нельзя.

Во втором примере $t = 1$, поэтому любое число компонент допустимо; максимальное число равно $n = 5$, что достигается удалением всех рёбер.

Задача D. Скука

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

Паук Иннокентий сидит на дереве и сучает. Ему настолько нечем заняться, что он готов делать что угодно. Например, просчитывать варианты плетения новой паутины.

Лес, где находится Иннокентий (как и любой другой), представляет собой набор деревьев (связных графов из n вершин и $n-1$ рёбер). В каждом дереве есть вершина, называемая *корнем*. Вершина u является *предком* вершины v , если они лежат в одном дереве и u лежит на пути от v до его корня. Вершина u называется *непосредственным предком* вершины v , если она является её предком и соединена с ней ребром.

Любая паутина представляет собой три вершины u, v, w , причудливо соединённые друг с другом. При этом вершина v должна быть предком вершин u и w , а расстояния между v и u и между v и w должны быть одинаковыми. Эти расстояния называются *высотой* паутины, вершины u, w называются *концами* паутины.

У Иннокентия есть q планов. Каждый план представляет собой вершину u и число h . Про такой план Иннокентий хочет знать, сколько существует различных паутин с концом в u , имеющих высоту h . Паутины считаются различными, если различаются наборы вершин, на которых они построены (т.е., в частности, $(1, 2, 3)$ и $(2, 3, 1)$ — одинаковые паутины). Помогите Иннокентию ответить на его вопросы.

Формат входных данных

В первой строке находится число n ($1 \leq n \leq 10^5$) — количество вершин в дереве.

Во второй строке находится n чисел $p_1, p_2, \dots, p_n$ ($0 \leq p_i \leq n$), где p_i — номер непосредственного предка вершины номер i или 0, если она является корнем своего дерева.

В третьей строке находится число q ($1 \leq q \leq 10^5$) — количество планов у паука.

В следующих q строках записаны по два числа u_j, h_j — параметры очередного плана ($1 \leq u_j, h_j \leq n$).

Формат выходных данных

Для каждого плана в отдельной строке выведите одно число — ответ на вопрос Иннокентия.

Пример

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

Задача Е. Паросочетание на дереве

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 1024 мегабайта

Вам дано дерево на n вершинах. Определим функцию $d(x, y)$, где x и y вершины дерева, как кратчайшее расстояние между этими вершинами. Вам необходимо найти такое множество $A = \{(x_1, y_1); \dots; (x_{\lfloor n/2 \rfloor}, y_{\lfloor n/2 \rfloor})\}$ размера $\lfloor n/2 \rfloor$, в котором каждая вершина встречается не более одного раза, и оно максимизирует $\sum_{i=1}^{\lfloor n/2 \rfloor} d(x_i, y_i)$.

Формат входных данных

На первой строке дано одно целое число $1 \leq n \leq 2 \cdot 10^5$. Далее идет $n - 1$ строка с описанием ребер в формате $1 \leq u_i < v_i \leq n$, где вершины u_i, v_i соединены ребром.

Формат выходных данных

Выведите искомое множество A , где на i -й строке написаны через пробел x_i и y_i . Если существует несколько оптимальных решений, то принимается любое.

Примеры

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

Задача F. Запросы на деревьях наносят ответный удар

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1.5 секунд
Ограничение по памяти: 256 мегабайт

Это ещё одна задача про запросы на дереве. Мы знаем, что вы не очень любите читать длинные и сложные легенды, поэтому опишем её кратко. Вам дано дерево на n вершинах, каждое ребро которого имеет целый положительный вес.

На этом дереве вам надо обрабатывать запросы двух типов:

1. Сделать вес i -го ребра равным w .
2. Посчитать расстояние от u до v .

Удачи, и да пребудет с вами сила.

Формат входных данных

В первой строке находится одно число n ($1 \leq n \leq 2 \cdot 10^5$) — количество вершин в дереве.

В следующих $n - 1$ строках находятся по три числа u_i, v_i, w_i ($1 \leq u_i, v_i \leq n, 1 \leq w_i \leq 10^9$) — концы и изначальный вес i -го ребра.

В следующей строке находится число q ($1 \leq q \leq 2 \cdot 10^5$) — количество запросов.

В следующих q строках даны запросы. Каждый запрос имеет один из двух видов:

- 1 i w — сделать вес i -го ребра равным w ($1 \leq i \leq n - 1, 1 \leq w \leq 10^9$). Рёбра нумеруются в том порядке, в котором они заданы во входных данных.
- 2 u v — вычислить расстояние от вершины с номером u до вершины с номером v ($1 \leq u, v \leq n$).

Формат выходных данных

Для каждого запроса второго типа в отдельной строке выведите одно число — ответ на этот запрос.

Пример

стандартный ввод	стандартный вывод
5	9
1 2 3	19
1 3 6	11
1 4 9	
4 5 10	
4	
2 2 3	
2 1 5	
1 3 1	
2 1 5	

Задача G. Деревянный патруль

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

Дано дерево на n вершинах.

Вы хотите поставить охранников в некоторые его вершины. Вершина считается защищенной, если есть охранник в ней или в ее соседе.

Для каждого $0 \leq k \leq n$ найдите количество способов расставить произвольное число охранников в некоторые вершины так, чтобы среди всех вершин дерева ровно k оказались защищенными. Ответ выведите по модулю $10^9 + 7$.

Формат входных данных

В первой строке ввода находится единственное число n — количество вершин в дереве ($1 \leq n \leq 2000$).

В i -й из следующих $n - 1$ строк находятся по два целых числа u_i, v_i ($1 \leq u_i, v_i \leq n$) — концы i -го ребра дерева.

Формат выходных данных

Выведите в единственной строке $n + 1$ число, i -е из которых является количеством способов расстановки для $k = i - 1$, взятым по модулю $10^9 + 7$.

Примеры

стандартный ввод	стандартный вывод
3 1 3 1 2	1 0 2 5
5 1 3 4 5 1 5 2 3	1 0 2 5 7 17

Задача Н. БЖД

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2.5 секунд
Ограничение по памяти:	512 мегабайт

В Берляндии есть N городов. Столица — город с номером 1. Все поезда Берляндских Железных Дорог (БЖД) следуют из столицы в другие города. Система железных дорог представляет собой дерево с корнем в столице. Как и на любой железной дороге, для разветвления путей используются стрелочные переводы: каждый город может перевести стрелку на любой перегон, идущий из него по направлению от столицы. После перевода стрелки проезд возможен только по этому перегону, а по остальным невозможен до тех пор, пока стрелка не будет переведена на них.

По БЖД будет совершено Q рейсов из столицы в различные города: i -й рейс совершается в город v_i . Если поезду необходимо проехать по перегону и стрелка не переведена на него, то стрелка переводится, а затем поезд продолжает движение. Для каждого рейса определите, сколько всего стрелок будет переключено за время поездки.

Если стрелка была переключена во время движения поезда, то она остается в этом состоянии, пока ее не переключат снова.

Формат входных данных

В первой строке вводится число N ($1 \leq N \leq 2 \cdot 10^5$) — количество городов в Берляндии.

В следующих N строках вводится схема БЖД: i -я строка схемы содержит описание перегонов, за которые отвечает город i . Первое число в описании k_i ($0 \leq k_i \leq N-1$) — число перегонов из города i , затем следует k_i чисел a_j ($1 \leq a_j \leq N$) — описание очередного перегона из города i . Изначально стрелка города i указывает на первый из перегонов — a_1 .

Гарантируется, что описанная схема перегонов является корневым деревом с корнем в столице.

В $N+2$ -й строке содержится число Q ($1 \leq Q \leq 2 \cdot 10^5$) — количество рейсов.

В следующих Q строках содержится по единственному числу v_i ($1 \leq v_i \leq N$) — номера городов, в которые из столицы отправляются рейсы.

Формат выходных данных

Выведите Q строк, в i -й строке выведите, сколько всего стрелок будет переведено для выполнения i -го рейса.

Система оценки

В данной задаче тесты разбиты на 6 подзадач, включая тесты из условия.

Подзадача	Баллы	Дополнительные ограничения
0	0	Тесты из условия
1	14	$N, Q \leq 100$
2	10	$N, Q \leq 2000$
3	20	$N, Q \leq 10^5$, схема БЖД имеет особый вид, см. ниже
4	28	$N, Q \leq 10^5$
5	28	Без дополнительных ограничений

В подзадаче 3 схема БЖД имеет следующий вид: $N = 2m$ четное число. Город с номером v , где $1 \leq v < m$, отвечает ровно за 2 перегона: в город $v+1$ и в город $v+m$. Город с номером m отвечает ровно за 1 перегон в город $2m$. Остальные города не отвечают ни за какие перегоны. Для лучшего понимания изучите второй тестовый пример и иллюстрацию к нему.

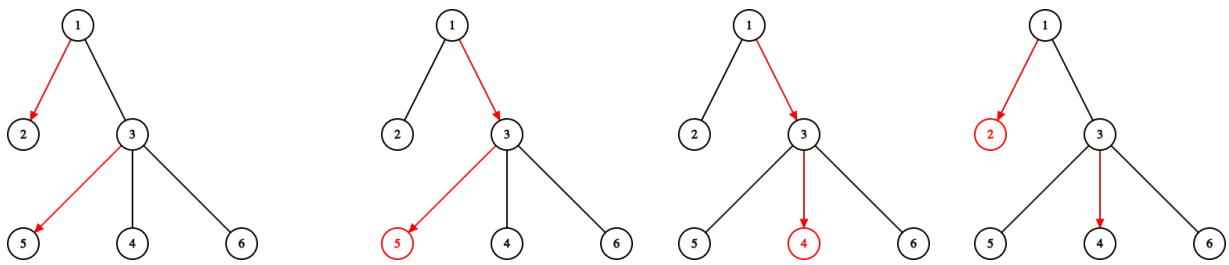
Оценка потестовая, вы набираете баллы за каждый пройденный тест.

Примеры

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

Замечание

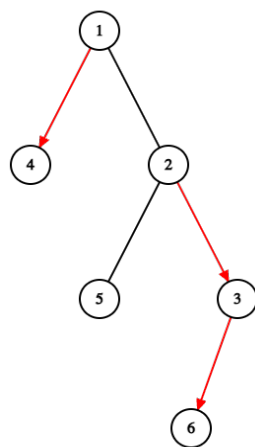
Иллюстрации к первому тестовому примеру (красные ребра обозначают направление стрелок, красная вершина обозначает, куда направляется поезд в этот день):



Изначальное состояние стрелок После первого дня После второго дня После третьего дня

В первый день требуется переключить только одну стрелку, $1 \rightarrow 2$ на $1 \rightarrow 3$. Во второй день тоже только одну стрелку, $3 \rightarrow 5$ на $3 \rightarrow 4$. И в третий день требуется переключить только стрелку $1 \rightarrow 3$ на $1 \rightarrow 2$.

Во втором тестовом примере изначальное положение стрелок выглядит следующим образом:



Задача I. Сумма на дереве

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

Вася очень любит деревья и суммы. На день рождения ему подарили новое большое дерево, в каждой вершине которого написано одно целое неотрицательное число. Вася сразу захотел посчитать такую сумму:

$$\sum_{u=1}^n \sum_{v=u}^n (f(u, v) - g(u, v)) \cdot \text{len}(u, v)$$

$f(u, v)$ — максимальное значение на пути от u до v , $g(u, v)$ — минимальное значение на пути от u до v , $\text{len}(u, v)$ — длина пути от u до v . К сожалению, дерево оказалось слишком большим, и Вася не успел закончить считать до отбоя. Ему очень хочется узнать ответ, помогите ему посчитать сумму!

Формат входных данных

В первой строке даётся одно число n ($1 \leq n \leq 5 \cdot 10^4$) — количество вершин дерева. Во второй строке даются n чисел a_i ($0 \leq a_i \leq 10^5$) — значения в вершинах дерева. В следующих $n - 1$ строках даётся описание рёбер дерева: по два числа u_i и v_i на строку — номера двух вершин, соединённых i -м ребром.

Формат выходных данных

Выведите одно число — искомую сумму.

Примеры

стандартный ввод	стандартный вывод
5 1 2 3 4 5 1 2 2 3 3 4 4 5	70
4 0 1 1 1 1 2 1 3 1 4	15

Задача J. Тайное общество

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

В одном городе действует тайное общество. Город состоит из n районов, соединённых $n-1$ дорогами. Между любыми двумя районами существует ровно один путь, то есть схема города представляет собой дерево.

В каждом районе живёт ровно один член общества.

Общество регулярно проводит встречи. Встреча важности x должна собрать ровно x членов общества. Для проведения такой встречи необходимо выбрать:

- район, в котором состоится встреча;
- ровно x членов общества, которые примут в ней участие.

Каждый участник добирается из своего района до района встречи по единственному простому пути. Стоимостью встречи называется суммарное количество дорог, пройденных всеми её участниками.

Иными словами, если встреча проходит в районе v , а множество участников равно S , то стоимость встречи равна $\sum_{u \in S} \text{dist}(u, v)$, где $\text{dist}(u, v)$ — количество дорог на пути между районами u и v .

Назовем *секретностью* встречи количество районов, стоимость проведения встречи в которых минимальна.

Для каждого x требуется посчитать максимально возможную *секретность* встречи важности x .

Формат входных данных

В первой строке дано целое число n — количество районов города ($1 \leq n \leq 2 \cdot 10^5$).

В следующих $n-1$ строках записаны по два целых числа a и b ($1 \leq a, b \leq n$) — номера районов, соединённых дорогой.

Гарантируется, что между любыми двумя районами существует ровно один путь.

Формат выходных данных

Выведите n целых чисел.

Число с номером x должно быть равно максимальной *секретности* по всем множествам размера x .

Числа можно вывести в одной строке или по одному в каждой строке.

Примеры

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

Замечание

В первом примере для встречи с важностью 2 оптимально отправить представителей из районов 1 и 5, тогда они могут выбрать районы 1, 2, 3, 5.