

## Вариант 7

### 1) #деревья, #Least Common Ancestor, #хэши, #дерево Фенвика

Как известно,  $X$  проводят всю свою жизнь в метро. Ведь только там можно так быстро передвигаться по городу, катаясь на поездах.

Из метро в город можно выбраться только с помощью  $n$  станций, расположенных в разных частях города. Некоторые станции соединены друг с другом тоннелями метро, по которым поезда могут передвигаться в обе стороны. Между каждой парой станций существует ровно один путь по тоннелям.

Когда  $X$  еще не знали карты метро, они постоянно забывали маршруты от одной станции до другой, и им приходилось спрашивать не дружелюбных работников метрополитена, в каком же направлении им двигаться, чтобы попасть на какую то станцию. Сотрудники метрополитена сообщали  $X$  название первой станции на пути от станции  $l$  к станции  $r$ .

Вскоре  $X$  устали от грубости сотрудников метро, и решили написать программу, которая по названиям станций  $l$  и  $r$  сообщает название первой станции на пути из  $l$  в  $r$ .

#### **Входные данные**

В первой строке  $n, m$  ( $0 < n, m < 10^5 + 1$ ) – количество станций и запросов соответственно.

В следующих  $n - 1$  строках заданы описания тоннелей  $a b$  – названия станций, строки из строчных символов латинского алфавита длиной не более 10.

Далее, описание запросов в формате  $l r$  – названия станций.

#### **Выходные данные**

Выведите для каждого запроса результат, в том порядке, в котором они даны во входных данных.

## 2) #деревья, #Least Common Ancestor, #дерево Фенвика

В корне одного дерева стоит муравей. Он видит, что в дереве  $n$  вершин, и они соединены  $n - 1$  ребрами так, что от любой вершины можно дойти до любой другой. Лист в дереве — это отличная от корня вершина, которая соединена ровно с одной другой вершиной.

Муравей хочет обойти все вершины дерева и вернуться назад к корню, пройдя по каждому ребру ровно два раза. При этом он хочет обойти все листья в заданном порядке. Ваша задача найти любой возможный путь муравья.

### Входные данные

В первой строке записано целое число  $n$  ( $3 \leq n \leq 30000$ ) — количество вершин в дереве. Далее следует  $n - 1$  строк — описания ребер. Каждое ребро описывается двумя числами — номерами вершин, которые оно соединяет. По каждому ребру можно идти в любом направлении. Вершины нумеруются с 1. Корень дерева имеет номер 1.

На следующей строке задан порядок обхода всех листьев дерева —  $k$  целых чисел, где  $k$  — количество листьев в дереве. Гарантируется, что этот порядок обхода содержит все листья дерева и только их ровно один раз.

### Выходные данные

Если искомого порядка обхода всех  $n$  вершин не существует, выведите  $-1$ . Иначе выведите  $2n - 1$  чисел — сам обход. Каждый раз, когда муравей приходит в вершину, выводите номер этой вершины.

### 3) #деревья, #дерево отрезков, #дерево Фенвика

У каждого сотрудника авиакомпании Oceanic Airlines, кроме директора, есть ровно один непосредственный начальник. Чтобы поощрять отличившихся сотрудников и целые отделы, директор авиакомпании может издавать два типа приказов:

1. «employee  $x$   $y$   $z$ » — если зарплата сотрудника  $x$  меньше  $y$  долларов, то повысить её на  $z$  долларов;
2. «department  $x$   $y$   $z$ » — если средняя зарплата в отделе под началом сотрудника  $x$  меньше  $y$  долларов, то повысить зарплату каждого сотрудника в этом отделе на  $z$  долларов (отдел включает самого сотрудника  $x$  и всех его подчинённых, не обязательно непосредственных).

Зная размер зарплаты всех сотрудников Oceanic Airlines на начало года и все приказы о её повышении, отданные директором в течение года, определите величину зарплат сотрудников на конец года. Можно считать, что в течение этого года авиакомпания не приняла на работу и не уволила ни одного сотрудника.

### Выходные данные

В первой строке записаны целые числа  $n$ ,  $q$  и  $s_0$  — количество сотрудников Oceanic Airlines, количество приказов о повышении зарплаты и зарплата директора на начало года ( $1 \leq n, q \leq 50\,000$ ;  $0 \leq s_0 \leq 10^9$ ). Сотрудники занумерованы целыми числами от 0 до  $n - 1$ , директор имеет номер ноль. Далее следует  $n - 1$  строка, в  $i$ -й из них записаны целые числа  $p_i$  и  $s_i$  — номер непосредственного начальника сотрудника с номером  $i$  и его зарплата на начало года ( $0 \leq p_i \leq i - 1$ ;  $0 \leq s_i \leq 10^9$ ). Далее идут  $q$  строк, описывающие приказы в том порядке, в котором их отдавал директор. Каждый приказ имеет вид «employee  $x$   $y$   $z$ » или «department  $x$   $y$   $z$ » (обозначения  $x$ ,  $y$ ,  $z$  описаны ранее).  $0 \leq x \leq n - 1$ ;  $1 \leq y, z \leq 10^9$ .

### Выходные данные

Перечислите зарплаты всех сотрудников Oceanic Airlines на конец года. Сотрудников следует описывать в порядке возрастания их номеров.

#### 4) #система непересекающихся множеств, #графы

Усатый-Полосатый XIII решил отомстить Ананси за освобождение бабочек, разрушив дом Ананси — его паутину. Паутина состоит из  $N$  узлов, некоторые из которых соединены нитями. Будем говорить, что два узла принадлежат одному кусочку, если от одного узла до другого можно добраться по нитям паутины. Усатый-Полосатый уже решил, какие нити и в каком порядке он будет рвать, и теперь хочет узнать, на сколько кусочков будет распадаться паутина после каждого из его действий.

#### **Входные данные**

В первой строке через пробел записаны числа  $N$  и  $M$  — количество узлов и нитей в паутине ( $2 \leq N \leq 100000; 1 \leq M \leq 100000$ ). В каждой из следующих  $M$  строк через пробел записаны два различных числа — номера узлов, которые соединяет очередная нить. Узлы занумерованы числами от 1 до  $N$ , нити занумерованы числами от 1 до  $M$  в том порядке, в котором они перечислены. Далее записано число  $Q$  — количество нитей, которое собирается порвать Усатый-Полосатый ( $1 \leq Q \leq M$ ). В последней строке записаны номера этих нитей — различные числа, отделяемые друг от друга пробелом.

#### **Выходные данные**

Выведите через пробел  $Q$  чисел — число кусочков, из которых будет состоять паутина Ананси после каждого обрыва нити.

5) #Least Common Ancestor, #дерево Фенвика, #деревья, #кратчайшие пути

Дано взвешенное дерево. Найти кратчайшее расстояние между заданными вершинами.

### **Входные данные**

Первая строка содержит целое число  $n$  — количество вершин в дереве ( $1 \leq n \leq 50000$ ).

Вершины нумеруются целыми числами от  $0$  до  $n - 1$ . В следующих  $n - 1$  строках содержится по три целых числа  $u, v, w$ , которые соответствуют ребру весом  $w$  ( $0 \leq w \leq 1000$ ), соединяющему вершины  $u$  и  $v$ . В следующей строке содержится целое число  $m$  — количество запросов ( $1 \leq m \leq 75000$ ). В следующих  $m$  строках содержится по два числа — номера вершин, расстояние между которыми необходимо вычислить.

### **Выходные данные**

Для каждого запроса выведите на отдельной строке одно число — искомое расстояние.

## 6) #бинарный поиск, #математика

Для заполнения бассейна имеются два крана — с горячей и с холодной водой. Холодная вода имеет температуру  $t_1$ , а горячая —  $t_2$ . Кран с холодной водой позволяет пропускать любое целое число литров воды в секунду в пределах от 0 до  $x_1$  включительно. Аналогично, кран с горячей водой может пропускать от 0 до  $x_2$  литров воды в секунду.

Если по первому крану течет  $y_1$  литров воды в секунду, а по второму —  $y_2$  литров, то результирующая температура воды в бассейне будет:

$$t = \frac{t_1 y_1 + t_2 y_2}{y_1 + y_2}$$

Саша хочет открыть оба крана таким образом, чтобы температура воды в бассейне была не меньше  $t_0$ , но при этом максимально близка к этому значению. Если оптимальных вариантов несколько, то Саша выбирает тот, при котором бассейн наполнится как можно быстрее.

Определите насколько нужно открыть каждый из кранов, чтобы Саша остался доволен.

### **Входные данные**

Даны пять целых чисел  $t_1$ ,  $t_2$ ,  $x_1$ ,  $x_2$  и  $t_0$  ( $1 \leq t_1 \leq t_0 \leq t_2 \leq 10^6$ ,  $1 \leq x_1, x_2 \leq 10^6$ ).

### **Выходные данные**

Выведите два целых числа  $y_1$  и  $y_2$ , разделенные пробелом ( $0 \leq y_1 \leq x_1$ ,  $0 \leq y_2 \leq x_2$ ).

## 7) #графы, #кратчайшие пути

Каждому бойцу 25-й стрелковой дивизии выдали новейшее средство связи — мобильный телеграф. С его помощью можно отправлять телеграммы командованию и боевым товарищам прямо на поле битвы. К сожалению, конструкция телеграфов ещё далека от совершенства — передавать сообщения можно только между некоторыми парами телеграфов.

Каждому устройству присвоен уникальный номер — строка из десяти десятичных цифр. С телеграфа  $a$  можно отправить сообщение на телеграф  $b$  только в том случае, если из номера  $a$  можно получить номер  $b$ , изменив в нём ровно одну цифру либо поменяв в нём две цифры местами. Время передачи сообщения с телеграфа  $a$  на телеграф  $b$  зависит от длины наибольшего общего префикса их номеров — чем больше его длина, тем быстрее передаётся сообщение.

Во время очередного сражения Анка из своей хорошо замаскированной позиции увидела небольшую группу белых, пытающуюся обойти обороняющихся красноармейцев с тыла. Какое минимальное время понадобится на доставку этой информации от Анки до Чапаева по телеграфу, возможно, с помощью других красноармейцев?

### Входные данные

В первой строке записано целое число  $n$  ( $2 \leq n \leq 50000$ ) — количество бойцов в дивизии. Во второй строке через пробел в порядке не возрастания записаны десять целых чисел в пределах от 1 до 10000 — время передачи сообщения с одного телеграфа на другой при длине общего префикса их номеров, равной нулю, единице, двум, ..., девяти. Далее идут  $n$  строк, содержащие номера телеграфов, выданных бойцам дивизии. Номер телеграфа Анки указан первым, а номер телеграфа Чапаева — последним. Все номера телеграфов попарно различны.

### Выходные данные

Если передать Чапаеву сообщение нельзя, выведите в единственной строке «-1». В противном случае в первой строке выведите минимальное время, за которое можно доставить сообщение. Во второй строке выведите количество бойцов, которые поучаствуют в его доставке, а в третьей строке выведите через пробел их номера в порядке от Анки к Чапаеву. Бойцы 25-й дивизии занумерованы числами от 1 до  $n$  в том порядке, в котором описаны номера их мобильных телеграфов на входе. Если существует несколько способов передать сообщение за минимальное время, выведите любой из них.

## 8) #графы, #циклы

В городе Фишбург есть несколько автобусных маршрутов. Никакие из них не имеют общего участка дороги, хотя возможны пересечения маршрутов и общие остановки. Старожилы Фишбурга утверждают, что возможно проехать с любой остановки на любую другую, возможно, с пересадками. Новый мэр решил реформировать транспортную систему города. Он предложил сделать только один маршрут, но проходящий через все отрезки дорог, по которым автобусы ездили ранее. Направление движения вдоль существовавших отрезков дорог должно остаться старым, также не должно использоваться дополнительных отрезков дорог.

Напишите программу, которая создаёт один из возможных новых маршрутов или находит, что такого маршрута создать нельзя.

### Входные данные

Первая строка ввода содержит количество старых маршрутов  $n$ . Каждая из следующих  $n$  строк содержит описание одного маршрута: количество остановок  $m$  и затем список этих остановок. Идентификаторы остановок — это положительные целые числа, не превосходящие  $10\,000$ . Маршрут представлен последовательностью из  $m + 1$  идентификатора остановок:  $l_1, l_2, \dots, l_m, l_{m+1} = l_1$ , которые последовательно посещаются автобусом, двигающимся вдоль этого маршрута. Маршрут может быть самопересекающимся. Маршрут всегда заканчивается той же остановкой, которой он начинается (все маршруты циклические).

Количество старых маршрутов:  $1 \leq n \leq 100$ .

Количество остановок:  $1 \leq m \leq 1000$ .

Идентификаторы остановок:  $1 \leq l_i \leq 10\,000$ .

### Выходные данные

Вывод содержит количество остановок в новом маршруте  $k$  и сам новый маршрут в том же формате, в каком маршруты даны во вводе. Последняя,  $(k+1)$ -я остановка должна совпадать с первой. Если новый маршрут невозможно проложить в соответствии с условием задачи, выведите одно число  $0$ .



## 9) #графы, #хэши, #поиск в глубину

Владислав Исенбаев — двукратный чемпион Урала по программированию, вице-чемпион TopCoder Open 2009, абсолютный чемпион ACM ICPC 2009. За то время, которое вы потратите на чтение этого условия, Владислав уже решил бы одну задачу. А может, и две...

Поскольку Владислав Исенбаев — выпускник СУНЦ УрГУ, неудивительно, что многие из бывших и действующих олимпиадников УрГУ знакомы с ним уже много лет. Некоторые из них с гордостью заявляют, что играли с Владиславом в одной команде. Или играли в команде с бывшими одноклассниками Владислава...

Определим *число Исенбаева* следующим образом. У самого Владислава это число равняется нулю. У тех, кто играл с ним в одной команде, оно равняется единице. У тех, кто играл вместе с одноклассниками Владислава, но не играл с ним самим, это число равняется двум, и так далее. Помогите автоматизировать процесс вычисления чисел Исенбаева, чтобы каждый олимпиадник в УрГУ мог знать, насколько близок он к чемпиону ACM ICPC.

### **Выходные данные**

В первой строке записано целое число  $n$  — количество команд ( $1 \leq n \leq 100$ ). В каждой из следующих  $n$  строк записаны составы этих команд в виде фамилий трёх участников. Фамилия каждого участника — непустая строка, состоящая из английских букв, длиной не более 20 символов. Первая буква фамилии — заглавная, все остальные — строчные. Фамилия Владислава — «Isenbaev».

### **Выходные данные**

Для каждого участника, представленного во входных данных, выведите в отдельной строке через пробел его фамилию и число Исенбаева. Если это число не определено, выведите вместо него «undefined». Участники должны быть упорядочены по фамилии в лексикографическом порядке.

## 10) #графы, #циклы

В состав Галактической империи входит  $N$  планет. Между большинством из них существуют гиперканалы. Новый император повелел, чтобы с любой планеты можно было попасть на любую другую, пройдя только через один гиперканал. Каналы устроены так, что позволяют путешествовать только в одну сторону.

Единственный оставшийся прокладчик гиперканалов расположен на базе около планеты с номером  $A$ . К сожалению, он не может путешествовать по уже существующим каналам, он всегда прокладывает новый. А наличие двух каналов в одном направлении между двумя планетами существенно осложняет навигацию. Ваша задача — найти такой маршрут для прокладчика, чтобы все необходимые каналы были построены, и не было бы построено лишних. В конце своего маршрута прокладчик должен оказаться на своей родной базе, с которой он начал движение.

### **Входные данные**

В первой строке находится число  $N \leq 1000$  и число  $A \leq N$ .  $N$  следующих строк содержит по  $N$  чисел — в  $i$ -й строке  $j$ -е число равно  $1$ , если есть канал от планеты  $i$  к планете  $j$ , иначе  $0$ . Известно, что Галактическая империя может удовлетворить свою потребность в гиперканалах прокладкой не более чем  $32000$  новых каналов.

### **Выходные данные**

Выведите последовательность, в которой следует прокладывать каналы. Каждая строка должна содержать два целых числа: номера планет, с какой и на какую следует проложить очередной гиперканал. Гиперканалы следует перечислить в порядке их прокладки. Гарантируется, что решение существует.