

## Задача А. Компоненты связности - 2

Имя входного файла: `stdin`  
Имя выходного файла: `stdout`  
Ограничение по времени: 3 секунды  
Ограничение по памяти: 1024 мегабайта

Дан неориентированный невзвешенный граф. Необходимо посчитать количество его компонент связности и вывести их.

### Формат входных данных

Во входном файле записано два числа  $N$  и  $M$  ( $0 < N \leq 100\,000$ ),  $0 \leq M \leq 100\,000$ ). В следующих  $M$  строках записаны по два числа  $i$  и  $j$  ( $1 \leq i, j \leq N$ ), которые означают, что вершины  $i$  и  $j$  соединены ребром.

### Формат выходных данных

В первой строчке выходного файла выведите количество компонент связности. Далее выведите сами компоненты связности в следующем формате: в первой строке количество вершин в компоненте, во второй — сами вершины в произвольном порядке.

### Примеры

stdin	stdout
6 4	3
3 1	3
1 2	1 2 3
5 4	2
2 3	4 5
	1
	6

## Задача В. Есть ли цикл?

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

Дан ориентированный граф. Требуется определить, есть ли в нем цикл.

### Формат входных данных

Сначала вводятся два числа:  $N$  и  $M$  ( $1 \leq N \leq 10^5, 0 \leq M \leq 10^5$ ) – количество вершин и ребер графа соответственно. Далее идет  $M$  пар чисел, задающих ребра.

### Формат выходных данных

Выведите «-1» без кавычек, если нет цикла. В противном случае выведите в первую строку количество вершин в цикле, а во вторую — номера вершин в цикле в порядке обхода. Если ответов несколько выведите любой.

### Примеры

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

## Задача С. Числа

Имя входного файла: `numbers.in`  
Имя выходного файла: `numbers.out`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 64 мегабайта

Витя хочет придумать новую игру с числами. В этой игре от игроков требуется преобразовывать четырехзначные числа не содержащие нулей при помощи следующего разрешенного набора действий:

1. Можно увеличить первую цифру числа на 1, если она не равна 9.
2. Можно уменьшить последнюю цифру на 1, если она не равна 1.
3. Можно циклически сдвинуть все цифры на одну вправо.
4. Можно циклически сдвинуть все цифры на одну влево.

Например, применяя эти правила к числу 1234, можно получить числа 2234, 1233, 4123 и 2341 соответственно.

Точные правила игры Витя пока не придумал, но пока его интересует вопрос, как получить из одного числа другое за минимальное количество операций.

### Формат входных данных

Во входном файле содержится два различных четырехзначных числа, каждое из которых не содержит нулей.

### Формат выходных данных

Программа должна вывести последовательность четырехзначных чисел, не содержащих нулей. Последовательность должна начинаться первым из данных чисел и заканчиваться вторым из данных чисел, каждое последующее число в последовательности должно быть получено из предыдущего числа применением одного из правил. Количество чисел в последовательности должно быть минимально возможным.

### Примеры

<code>numbers.in</code>	<code>numbers.out</code>
9876	9876
8876	8769
	8768
	8876

## Задача D. Красно-синий граф

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

Даны  $N$  точек, занумерованных числами  $1, 2, \dots, N$ . От каждой точки с меньшим номером к каждой точке с большим номером ведет стрелка красного или синего цвета. Раскраска стрелок называется однотонной, если нет двух таких точек  $A$  и  $B$ , что от  $A$  до  $B$  можно добраться как только по красным стрелкам, так и только по синим.

Ваша задача — по заданной раскраске определить, является ли она однотонной.

### Формат входных данных

В первой строке записано единственное число  $N$  ( $3 \leq N \leq 5000$ ).

В следующих  $N - 1$  строках идет описание раскраски. В  $(i + 1)$ -й строке записано  $(N - i)$  символов R (красный) или B (синий), соответствующих цвету стрелок, выходящих из точки  $i$  и входящих в точки  $(i + 1), (i + 2), \dots, N$  соответственно.

### Формат выходных данных

Выведите YES, если приведенная раскраска является однотонной, и NO в противном случае.

### Примеры

стандартный ввод	стандартный вывод
3 RB R	NO
3 RR R	YES

## Задача Е. Два автобуса

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

Начинается осенняя смена по информатике в образовательном центре Сириус. Все участники смены благополучно прилетели, и из аэропорта до центра их повезут на двух больших автобусах.

Организаторы смены решили заранее распределить участников по автобусам, чтобы ускорить процесс посадки. Но вскоре оказалось, что при распределении детей надо учитывать некоторые факторы. Из некоторых источников организаторы знают, что некоторые дети дружат, и они расстроятся, если посадить их в разные автобусы. И наоборот, некоторые дети недолюбливают друг друга, и сажать их в один автобус нельзя.

Учесть все эти ограничения достаточно сложно, поэтому организаторы попросили вас написать программу, которая получит корректное распределение участников по двум автобусам. Автобусы достаточно большие, чтобы вместить любое количество детей.

### Формат входных данных

В первой строке заданы 3 числа  $n, m, k$  — число участников смены, количество пар участников которые должны оказаться в одном автобусе и количество пар участников, которые должны оказаться в разных автобусах ( $1 \leq n \leq 10^5, 0 \leq m, k \leq 10^5$ ).

В следующих  $m$  строках перечислены пары номеров участников  $u_i$  и  $v_i$ , которые должны сидеть в одном автобусе ( $1 \leq u_i, v_i \leq n, u_i \neq v_i$ ).

В следующих  $k$  строках перечислены пары номеров участников  $p_i$  и  $t_i$ , которые должны сидеть в разных автобусах ( $1 \leq p_i, t_i \leq n, p_i \neq t_i$ ).

### Формат выходных данных

Если требуемого распределения не существует, в единственной строке выведите «No» (без кавычек).

Иначе в первой строке выведите «Yes». Во второй строке выведите количество детей в одном автобусе, а затем их номера через пробел. В третьей строке выведите количество детей в другом автобусе, и их номера через пробел.

### Примеры

стандартный ввод	стандартный вывод
3 1 1 1 2 2 3	Yes 2 1 2 1 3
3 0 3 1 2 2 3 1 3	No

## Задача F. Производство деталей

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

Предприятие «Авто-2010» выпускает двигатели для известных во всём мире автомобилей. Двигатель состоит ровно из  $n$  деталей, пронумерованных от 1 до  $n$ , при этом деталь с номером  $i$  изготавливается за  $p_i$  секунд. Специфика предприятия «Авто-2010» заключается в том, что там одновременно может изготавливаться лишь одна деталь двигателя. Для производства некоторых деталей необходимо иметь предварительно изготовленный набор других деталей.

Генеральный директор «Авто-2010» поставил перед предприятием амбициозную задачу — за наименьшее время изготовить деталь с номером 1, чтобы представить её на выставке.

Требуется написать программу, которая по заданным зависимостям порядка производства между деталями найдёт наименьшее время, за которое можно произвести деталь с номером 1.

### Формат входных данных

Первая строка входного файла содержит число  $n$  ( $1 \leq n \leq 10^5$ ) — количество деталей двигателя.

Вторая строка содержит  $n$  натуральных чисел  $p_1, p_2, \dots, p_n$ , определяющих время изготовления каждой детали в секундах. Время для изготовления каждой детали не превосходит  $10^9$  секунд.

Каждая из последующих  $n$  строк входного файла описывает характеристики производства деталей. Здесь  $i$ -я строка содержит число деталей  $k_i$ , которые требуются для производства детали с номером  $i$ , а также их номера. В  $i$ -й строке нет повторяющихся номеров деталей. Сумма всех чисел  $k_i$  не превосходит  $2 \cdot 10^5$ .

Известно, что не существует циклических зависимостей в производстве деталей.

### Формат выходных данных

В первой строке выходного файла должны содержаться два числа: минимальное время (в секундах), необходимое для скорейшего производства детали с номером 1 и число  $k$  деталей, которые необходимо для этого произвести.

Во второй строке требуется вывести через пробел  $k$  чисел — номера деталей в том порядке, в котором следует их производить для скорейшего производства детали с номером 1.

### Примеры

стандартный ввод	стандартный вывод
3 100 200 300 1 2 0 2 2 1	300 2 2 1
2 2 3 1 2 0	5 2 2 1
4 2 3 4 5 2 3 2 1 3 0 2 1 3	9 3 3 2 1



Путем в дереве называется последовательность вершин, в которой две соседние вершины соединены ребром. Все вершины в пути должны быть различны.

Для того, чтобы оценить красоту фейерверка, необходимо определить, какое максимальное количество вершин может содержать путь в дереве, которым представляется фейерверк. На рис. 3 приведен путь в дереве  $T^2$ , содержащий максимальное количество вершин. Таким образом, красота фейерверка с базовым деревом  $T$  и мощностью 2 равна 10.

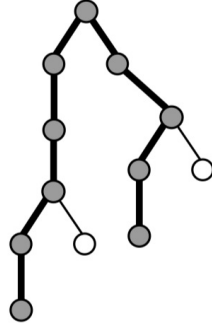


Рис. 3: Путь в дереве  $T^2$ , содержащий максимальное количество вершин

Требуется написать программу, которая по описанию дерева  $T$  и натуральному числу  $m$  определяет красоту фейерверка с базовым деревом  $T$  и мощностью  $m$ .

### Формат входных данных

В первой строке задано число  $n$  ( $3 \leq n \leq 200\,000$ ) и число  $m$  ( $1 \leq m \leq 200\,00$ ) — размер дерева и степень, в которую его требуется возвести соответственно.

Дальше следует строка, содержащая  $n - 1$  число:  $p_2, \dots, p_n$  — предки соответствующих вершин дерева,  $1 \leq p_i < i$

Корнем дерева является вершина с номером 1, гарантируется, что она не является листом.

### Формат выходных данных

Выведите одно число — число вершин в диаметре дерева.

### Примеры

стандартный ввод	стандартный вывод
4 2 1 1 2	10

## Задача Н. Наводнение

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

На базе «Берендеевы поляны» все домики соединены между собой дорожками. Каждая дорожка имеет свою высоту над уровнем озера Лель. Каждый домик может быть соединён с другим несколькими дорожками, но дорожек, ведущих из домика в него же, нет.

После сильных дождей озеро выходит из берегов и затапливает дорожки. Дорожка затапливается в том случае, если уровень воды больше или равен её высоте. Поскольку в лагере есть только самокаты и велосипеды, а лодок ещё не закупили, для нормального функционирования лагеря от любого домика до любого другого домика должен существовать путь по незатопленным дорожкам, возможно, по нескольким. Помогите сотрудникам базы определить минимальный уровень воды, при котором найдутся такие два домика, что от одного из них никак нельзя добраться до другого. Гарантируется, что до начала сильных дождей из любого домика можно было попасть в любой другой.

### Формат входных данных

В первой строке даны два числа  $A$  и  $B$  — количество домиков и дорожек соответственно ( $2 \leq A \leq 10\,000$ ,  $1 \leq B \leq 20\,000$ ). Следующие  $B$  строк содержат тройки чисел  $(s_i, f_i, h_i)$ , где  $s_i$  и  $f_i$  — номера домиков, которые соединяет  $i$ -я дорожка, а  $h_i$  — её высота над уровнем озера в миллиметрах ( $1 \leq s_i \leq A$ ,  $1 \leq f_i \leq A$ ,  $1 \leq h_i \leq 1\,000\,000$ ).

### Формат выходных данных

Выведите минимальную высоту, на которую должна подняться вода, чтобы в лагере нашлось хотя бы два домика, между которыми нельзя пройти по незатопленным дорожкам.

### Примеры

стандартный ввод	стандартный вывод
2 1 1 2 100	100
4 5 1 2 100 1 3 400 2 3 300 2 4 200 3 4 500	300

## Задача I. Доставка кефирчика

Имя входного файла: kefir.in  
Имя выходного файла: kefir.out  
Ограничение по времени: 3 секунды  
Ограничение по памяти: 64 мегабайта

Во время проведения очередной Межгалактической Летней Компьютерной Школы (МЛКШ) организаторы столкнулись с проблемой доставки кефирчика для вечерки. Дело в том, что кефирчик производят на планете под номером 1, а сами школьники живут на планете  $n$ , поэтому на доставку кефирчика тратится довольно большое время, а значит он успеваает испортиться.

К счастью, галактическая транспортная система «Берендеев-Экспресс» постепенно внедряет новые кефиропроводы, способные передавать кефир со скоростью, в два раза превышающей скорость старых моделей. А именно, с любой планеты на любую по старым кефиропроводам кефир проходит за два года, а по новым — за один.

Разумеется, грешно было бы не воспользоваться инновационными технологиями, поэтому директор МЛКШ попросил вас написать программу, которая по данным о имеющихся кефиропроводах (как новых, так и старых) узнает кратчайший путь от планеты 1 до планеты  $n$ .

### Формат входных данных

В первой строке входного файла даны два целых числа  $n$  и  $m$  ( $1 \leq n \leq 100\,000$ ,  $0 \leq m \leq 100\,000$ ) — количество планет и количество кефиропроводов соответственно. В последующих  $m$  строках даны тройки натуральных чисел  $u_i$ ,  $v_i$  и  $c_i$ . Числа  $u_i$  и  $v_i$  обозначают номера планет, соединенных  $i$ -м кефиропроводом, а  $c_i$  ( $c_i = 1$  или  $c_i = 2$ ) — количество лет, которое потребуется, чтобы передать кефир с одной планеты на другую через  $i$ -й кефиропровод. Планеты во входном файле нумеруются с единицы. Кефир по трубопроводам можно передавать в обоих направлениях.

### Формат выходных данных

В выходной файл требуется вывести одно число — количество лет, которое требуется, чтобы доставить кефир с планеты 1 на планету  $n$ . Если доставка невозможна, то в выходной файл требуется вывести «-1».

### Примеры

kefir.in	kefir.out
3 2 1 2 2 2 3 1	3
3 1 2 3 1	-1
2 5 1 2 1 1 2 2 1 2 1 1 1 2 2 2 1	1