

Задача А. От списка ребер к матрице смежности, ориентированный вариант

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

Простой ориентированный граф задан списком ребер, выведите его представление в виде матрицы смежности.

Формат входных данных

На вход программы поступают числа n ($1 \leq n \leq 100$) — количество вершин в графе и m ($1 \leq m \leq n(n-1)$) — количество ребер. Затем следует m пар чисел — ребра графа.

Формат выходных данных

Выведите матрицу смежности заданного графа.

Пример

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

Задача В. Степени вершин по списку ребер

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

Неориентированный граф задан списком ребер. Найдите степени всех вершин графа.

Формат входных данных

Сначала вводятся числа n ($1 \leq n \leq 100$) — количество вершин в графе и m ($1 \leq m \leq \frac{n(n-1)}{2}$) — количество ребер. Затем следует m пар чисел — ребра графа. Каждая вершина — число от 1 до n .

Формат выходных данных

Выведите n чисел — степени вершин графа. i -ое выведенное число должно быть равно степени вершины с номером i .

Пример

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

Задача С. Обход графа

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

Дан неориентированный невзвешенный граф. Для него вам необходимо найти количество вершин, лежащих в одной компоненте связности с данной вершиной (считая эту вершину).

Формат входных данных

В первой строке входных данных содержатся два числа: N и S ($1 \leq N \leq 100$; $1 \leq S \leq N$), где N — количество вершин графа, а S — заданная вершина. В следующих N строках записано по N чисел — матрица смежности графа, в которой 0 означает отсутствие ребра между вершинами, а 1 — его наличие. Гарантируется, что на главной диагонали матрицы всегда стоят нули.

Формат выходных данных

Выведите одно целое число — искомое количество вершин.

Пример

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

Задача D. Дерево?

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

Имеется неориентированный граф, состоящий из N вершин и M ребер. Необходимо проверить, является ли граф деревом. Напомним, что дерево — это связный граф, в котором нет циклов (следовательно, между любой парой вершин существует ровно один простой путь). Граф называется связным, если от одной вершины существует путь до любой другой.

Формат входных данных

Во входном файле в первой строке содержатся два целых числа N и M ($1 \leq N \leq 100$, $0 \leq M \leq 1000$), записанные через пробел. Далее следуют M различных строк с описаниями ребер, каждая из которых содержит два натуральных числа A_i и B_i ($1 \leq A_i, B_i \leq N$), где A_i и B_i — номера вершин, соединенных i -м ребром.

Формат выходных данных

В выходной файл выведите слово «YES», если граф является деревом, или «NO» в противном случае.

Примеры

стандартный ввод	стандартный вывод
3 2 1 2 1 3	YES
4 3 1 2 1 3 2 3	NO

Задача Е. Компоненты связности

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

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 ($0 < N \leq 100000, 0 \leq M \leq 100000$). 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 отсортированном порядке.

Пример

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

Задача F. Долой списывание!

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта

Во время теста Михаил Дмитриевич заметил, что некоторые лкшат обмениваются записками. Сначала он хотел поставить им всем двойки, но в тот день Михаил Дмитриевич был добрым, а потому решил разделить лкшат на две группы: списывающих и дающих списывать, и поставить двойки только первым.

У Михаила Дмитриевича записаны все пары лкшат, обменявшихся записками. Требуется определить, сможет ли он разделить лкшат на две группы так, чтобы любой обмен записками осуществлялся от лкшонка одной группы лкшонку другой группы.

Формат входных данных

В первой строке находятся два числа N и M — количество лкшат и количество пар лкшат, обменивающихся записками ($1 \leq N \leq 100$, $0 \leq M \leq \frac{N(N-1)}{2}$). Далее в M строках расположены описания пар лкшат: два различных числа, соответствующие номерам лкшат, обменивающихся записками (нумерация лкшат идёт с 1). Каждая пара лкшат перечислена не более одного раза.

Формат выходных данных

Необходимо вывести ответ на задачу Павла Олеговича. Если возможно разделить лкшат на две группы, выведите «YES»; иначе выведите «NO».

Пример

стандартный ввод	стандартный вывод
3 2 1 2 2 3	YES

Задача G. 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

В первой строке вводится число n - количество вершин и m - количество ребер. ($1 \leq n, m \leq 10^5$).
Далее в m строках следует по 2 числа u, v - вершины графа, соединенные ребром.

Формат выходных данных

Выведите 0, если в заданном графе нет цикла, и 1, если он есть.

Примеры

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

Задача Н. Тесей и минотавр

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

Вы — Тесей, герой мифов и легенд. Сейчас вы готовитесь к победе над минотавром. Вскоре Вам предстоит зайти в лабиринт и сразится с древним чудовищем.

Лабиринт представляет из себя n комнат. Вы подкупили одного из строителей лабиринта. Теперь вы знаете, что комнаты лабиринта соединены $n - 1$ тоннелями. За дополнительную плату вы получили схему лабиринта.

Так как ваш отец — царь Афин, то денег на подкупы у вас много. Поэтому вы также подкупили человека, который знает повадки минотавра. Так вы узнали, что минотавр — расчетливое создание: он хочет, чтобы люди, которых запускают в лабиринт, устали перед встречей с ним. Поэтому он выбирает для логова комнату, которая наиболее удалена от входной комнаты.

Вы спешите на следующие подвиги, поэтому вы не хотите тратить лишнее время на путь до логова минотавра. Поэтому вы должны точно определить комнату, в которой находится логово минотавра.

Формат входных данных

В первой строке входных данных вы получаете число $n \leq 10^5$ — количество комнат и число $1 \leq s \leq n$ — номер входной комнаты.

В последующих $n - 1$ строках находятся числа x и y : $1 \leq x, y \leq n$ — номера комнат между которыми существует тоннель.

Формат выходных данных

В единственной строке выходных данных выведите несколько чисел — все номера комнат в порядке возрастания, в которых может располагаться логово минотавра.

Примеры

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

Задача I. Прогулочный маршрут

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

Петя успешно сдал вступительные экзамены в классы «Силаэдр». Это радостное событие он решил отметить прогулкой по Москве. Как известно, этот красивый город состоит из улочек-дорожек, на каждую из которых можно попасть ровно с одной другой. День был солнечным, и Петя захотел пройтись от 1 улочки до n -й. К сожалению, несмотря на его ум, мальчик был ещё маленьким, и не умел ориентироваться по карте, поэтому попросил Вас подсказать нужный ему путь. Помогите Пете, чтобы он остался в живых и не потерялся.

Формат входных данных

В первой строке записано целое число n ($2 \leq n \leq 200000$) — количество дорожек. Далее во второй строке записано $n - 1$ целое число p_i , где p_i равно номеру дорожки, с которой можно попасть на дорожку с номером i для $i = 2, 3, \dots, n$.

Формат выходных данных

Выведите путь от 1 дорожки до n -й.

Примеры

стандартный ввод	стандартный вывод
8 1 1 2 2 3 2 5	1 2 5 8
6 1 2 3 4 5	1 2 3 4 5 6

Задача J. Найди Ереван на карте

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

Город Ереван — центр человеческой цивилизации

Таксист с площади Сустанси Давида

Сегодня Вам предстоит найти центр мира нашего — город Ереван. Вам задана карта рейсов между городами мира — граф G , состоящий из n вершин и m ребер.

Известно, что Ереван, как центр мира, сильно связан со всеми остальными городами. Конкретнее, из Еревана можно добраться до любого другого города, совершив не более двух перелётов. Найдите на карте мира город Ереван. Гарантируется, что Ереван в мире ровно один.

Формат входных данных

В первой строке вводятся два числа $1 \leq n \leq 2000, 1 \leq m \leq 10^5$ — количество вершин и ребер в графе соответственно.

В последующих m строках вводятся по два числа $1 \leq v_i, u_i \leq n$ — ребра графа.

Формат выходных данных

Выведите единственное число — номер вершины, соответствующей Еревану.

Примеры

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

Задача К. Свинки-копилки

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

У Васи есть n свинок-копилков, свинки занумерованы числами от 1 до n . Каждая копилка может быть открыта единственным соответствующим ей ключом или разбита.

Вася положил ключи в некоторые из копилков (он помнит, какой ключ лежит в какой из копилков). Теперь Вася собрался купить машину, а для этого ему нужно достать деньги из всех копилков. При этом он хочет разбить как можно меньшее количество копилков (ведь ему еще нужно копить деньги на квартиру, дачу, вертолет. . .). Помогите Васе определить, какое минимальное количество копилков нужно разбить.

Формат входных данных

В первой строке содержится число n — количество свинок-копилков ($1 \leq n \leq 100$). Далее идет n строк с описанием того, где лежит ключ от какой копилки: в i -й из этих строк записан номер копилки, в которой находится ключ от i -й копилки.

Формат выходных данных

Выведите единственное число: минимальное количество копилков, которые необходимо разбить.

Пример

стандартный ввод	стандартный вывод
4	2
2	
1	
2	
4	

Задача L. Удаление клеток

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

Из прямоугольного листа клетчатой бумаги (m строк, m столбцов) удалили некоторые клетки. На сколько кусков распадётся оставшаяся часть листа? Две клетки не распадаются, если они имеют общую сторону.

Формат входных данных

В первой строке находятся числа n и m ($1 \leq n \cdot m \leq 10^4$), в следующих n строках - по m символов. Если клетка не была вырезана, этому соответствует знак #, если вырезана - точка.

Формат выходных данных

Вывести одно число.

Пример

стандартный ввод	стандартный вывод
5 10 ##.#####. .###.#.... ###.###.#. ..##.....# .###.#####	5

Задача М. Получи дерево

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

Дан связный неориентированный граф без петель и кратных ребер. Разрешается удалять из него ребра. Требуется получить дерево.

Формат входных данных

Сначала вводятся два числа от 1 до 10^5 : n, m – количество вершин и ребер графа соответственно. Далее идет m пар чисел, задающих ребра. Гарантируется, что граф связный.

Формат выходных данных

Выведите $n - 1$ пару чисел — ребра, которые войдут в дерево. Ребра можно выводить в любом порядке.

Пример

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

Задача N. Таможенные пошлины

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

Недавно королева страны AlgoLand придумала новый способ отмывания денег для своего королевского двора. Она решила, что всякий житель, желающий совершить путешествие из одного города страны в другой, должен расплатиться за это желание своими деньгами.

В стране AlgoLand есть N городов, пронумерованных от 1 до N . Некоторые города соединены дорогами, движение по которым разрешено в двух направлениях. Начиная движение по какой-нибудь дороге, путешественник обязательно должен доехать до ее конца.

Предположим теперь, что житель страны хочет совершить путешествие из города A в город B . Новый указ королевы гласит, что при проезде по любой дороге страны во время этого путешествия, полицейские могут взять с этого жителя таможенную пошлину в пользу королевского двора (а могут и не взять). Если при этом у жителя недостаточно денег для уплаты пошлины, то он автоматически попадает в тюрьму. Указ также устанавливает величину пошлины для каждой дороги страны. Так как королева заботится о жителях своей страны, то она запретила полицейским брать с жителя пошлину более чем три раза во время одного путешествия.

Отметим, что если существует несколько способов попасть из города A в город B , то житель может выбрать для путешествия любой из них по собственному желанию.

Напишите программу, которая определяет, какую минимальную сумму денег должен взять с собой житель, чтобы гарантированно не попасть в тюрьму во время путешествия.

Формат входных данных

Первая строка входного файла содержит числа N и M ($2 \leq N \leq 10\,000$, $1 \leq M \leq 100\,000$), разделенные пробелом — количества городов и дорог. Следующие M строк описывают дороги. Каждая из этих строк описывает одну дорогу и содержит три числа X , Y , Z ($1 \leq X, Y \leq N$; $X \neq Y$; $1 \leq Z \leq 1\,000\,000\,000$), разделенных пробелами, означающие, что дорога соединяет города X и Y и пошлина за проезд по ней равна Z денежных единиц. Все числа Z целые. Последняя строка содержит числа A и B ($1 \leq A, B \leq N$; $A \neq B$) — номера начального и конечного городов путешествия. Гарантируется, что существует хотя бы один способ проезда из A в B .

Формат выходных данных

Единственная строка выходного файла должна содержать одно число, равное минимальной сумме денег, которую должен взять с собой житель, чтобы иметь возможность совершить путешествие из города A в город B и при этом гарантированно не попасть в тюрьму независимо от действий полицейских.

Пример

стандартный ввод	стандартный вывод
5 6 1 2 10 1 3 4 3 2 3 1 4 1 4 5 2 5 2 3 1 2	6

Задача О. 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 ($1 \leq n \leq 100\,000$) — 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 200 000.

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