

А. Мосты и компоненты

ограничение по времени на тест: 2 секунды
ограничение по памяти на тест: 256 мегабайт
ввод: стандартный ввод
вывод: стандартный вывод

Дан неориентированный граф (не обязательно связный). Граф может содержать петли и кратные ребра.

Выведите все компоненты реберной двусвязности графа (максимальные подмножества вершин, такие что подграф на них не теряет связность при удалении любого ребра).

Входные данные

Первая строка содержит числа n и m ($1 \leq n \leq 100\,000$, $0 \leq m \leq 100\,000$) — количество вершин и ребер в графе.

Следующие m строк задают ребра графа.

Выходные данные

В первой строке выведите количество компонент, в следующих за ней строках выведите сами компоненты, по одной на строку.

Вершины в каждой компоненте должны идти в возрастающем порядке, компоненты нужно вывести в лексикографическом порядке.

Компонента — вектор номеров своих вершин. Лексикографически сравниваются вектора.

В. Unique Topsort

ограничение по времени на тест: 1 секунда
ограничение по памяти на тест: 256 мегабайт
ввод: стандартный ввод
вывод: стандартный вывод

Дан ориентированный ациклический граф G . Проверить, что существует единственный топологический порядок вершин графа.

Входные данные

Первая строка входных данных содержит число вершин графа n ($1 \leq n \leq 100\,000$) и число ребер графа m ($0 \leq m \leq 100\,000$). Следующие m строк содержат пары чисел от 1 до n , задающие начало и конец соответствующего ребра. Гарантируется, что граф не содержит циклов.

Выходные данные

Если топологический порядок единственный, выведите на первой строке YES , а на второй номера вершин в топологическом порядке, иначе выведите NO .

Е. Идеальный путь

ограничение по времени на тест: 2 секунды
ограничение по памяти на тест: 256 мегабайт
ввод: стандартный ввод
вывод: стандартный вывод

Есть лабиринт развлечений из n комнат и m цветных переходов между комнатами. Участник изначально попадает в комнату 1, цель — добраться до комнаты n . Несколько участников одновременно стартуют из комнаты 1. Каждый участник пройдет некоторым путем из комнаты 1 в комнату n , записывая цвета переходов, по которым прошел. Выигрывает участник с кратчайшей последовательностью, а если таких несколько, с лексикографически меньшей среди кратчайших. Эндрю тоже участвует и очень хочет выиграть. Помогите ему, найдите кратчайшую, а из таких лексикографически минимальную последовательность переходов из 1 в n .

Примечание

Последовательность (a_1, a_2, \dots, a_k) лексикографически меньше последовательности (b_1, b_2, \dots, b_k) если существует i такое, что $a_i < b_i$ и $a_j = b_j$ для всех $j < i$.

Входные данные

Первая строка содержит целые числа n и m — количество комнат и переходов соответственно ($2 \leq n \leq 100\,000$, $1 \leq m \leq 200\,000$). Следующие m строк содержат описания переходов. Каждый переход задается тремя целыми числами: a_i , b_i и c_i — номера комнат, которые он соединяет, и цвет перехода ($1 \leq a_i, b_i \leq n$, $1 \leq c_i \leq 10^9$). Каждый переход может быть использован в обоих направлениях. Две комнаты могут быть соединены несколькими переходами, также может быть переход из комнаты в саму себя. Гарантируется, что из комнаты 1 можно как-нибудь попасть в комнату n .

Выходные данные

На первой строке выведите k — длину кратчайшей последовательности переходов из комнаты 1 в комнату n . На следующей строке k чисел — цвета переходов в порядке их прохождения.

Ф. Кратчайший путь коня

ограничение по времени на тест: 1 секунда
ограничение по памяти на тест: 64 мегабайта
ввод: стандартный ввод
вывод: стандартный вывод

На шахматной доске размером 8×8 заданы две клетки. Соедините эти клетки кратчайшим путем коня.

Входные данные

Программа получает на вход координаты двух клеток, каждая в отдельной строке. Координаты клеток задаются в виде буквы (от "a" до "h") и цифры (от 1 до 8) без пробелов.

Выходные данные

Программа должна вывести путь коня, начинающийся и заканчивающийся в данных клетках и содержащий наименьшее число клеток.

Г. Островные государства

ограничение по времени на тест: 2 секунды
ограничение по памяти на тест: 256 мегабайт
ввод: стандартный ввод
вывод: стандартный вывод

Некогда великая островная страна Байтландия переживает суровые феодальные времена. За главенство над островом борются два самых влиятельных барона. Каждый город страны контролируется одним из правителей. Как водится издревле, некоторые из городов соединены двухсторонними дорогами. Бароны очень не любят друг друга и стараются устроить сопернику как можно больше пакостей. В частности, для того, чтобы пройти по дороге, соединяющей города различных правителей, надо заплатить пошлину — один байтландский рубль. Кроме того, за выезд из городов с четными номерами берется удвоенная пошлина.

Программист Вася живет в городе номер 1. С наступлением лета он собирается съездить в город N на Всебайтландское собрание программистов. Разумеется, он хочет потратить при этом как можно меньше денег и помочь ему здесь, как обычно, предлагается Вам.

Входные данные

В первой строке входного файла записаны два числа N и M ($2 \leq N \leq 100\,000$, $1 \leq M \leq 100\,000$) — количество городов и количество дорог соответственно.

В следующей строке содержится информация о городах — N чисел 1 или 2 — какому из баронов принадлежит соответствующий город.

В последних M строках записаны пары $1 \leq a, b \leq N$, $a \neq b$. Каждая пара означает наличие дороги из города a в город b . По дорогам Байтландии можно двигаться в любом направлении.

Выходные данные

Если искомого пути не существует, выведите единственное слово `impossible`. В противном случае в первой строке напишите минимальную стоимость и количество посещенных городов, а во вторую выведите эти города в порядке посещения. Если минимальных путей несколько, выведите любой.

Н. Транзитивное замыкание

ограничение по времени на тест: 1 секунда
ограничение по памяти на тест: 256 мегабайт
ввод: стандартный ввод
вывод: стандартный вывод

Дан ориентированный граф. Найдите его транзитивное замыкание, то есть для каждой пары вершин a, b определите, есть ли путь из a в b .

Входные данные

На первой строке число вершин n ($1 \leq n \leq 1\,000$). Следующие n строк имеют длину n , состоят из нулей и единиц и задают матрицу смежности графа. Единица в i -й строке, j -м столбце обозначает ребро из i в j .

Выходные данные

Выведите матрицу смежности транзитивного замыкания данного графа.

К. Флойд

ограничение по времени на тест: 2 секунды
ограничение по памяти на тест: 256 мегабайт
ввод: стандартный ввод
вывод: стандартный вывод

Полный ориентированный взвешенный граф задан матрицей смежности. Постройте матрицу кратчайших путей между его вершинами. Гарантируется, что в графе нет циклов отрицательного веса.

Входные данные

В первой строке вводится единственное число N ($1 \leq N \leq 100$) — количество вершин графа. В следующих N строках по N чисел задается матрица смежности графа (j -е число в i -й строке — вес ребра из вершины i в вершину j). Все числа по модулю не превышают 100. На главной диагонали матрицы — всегда нули.

Выходные данные

Выведите N строк по N чисел — матрицу расстояний между парами вершин, где j -е число в i -й строке равно весу кратчайшего пути из вершины i в j .