

A. Topological sort

2 seconds, 256 megabytes

You are given a directed graph. Find its topological sorting.

Input

The first line contains two integers n and m ($1 \leq n \leq 100\,000$, $m \leq 100\,000$) — the number of vertices and the number of edges in the graph, respectively.

Next m lines describe edges of the graph. Each line contains two integers v and u ($1 \leq v, u \leq n$; $v \neq u$) — describing the edge starting at v and ending at u .

Output

If no topological sorting exists, output "-1".

Otherwise, output the sequence of vertices which describes the topological ordering. If several orderings exist, output any.

input

```
6 6
1 2
3 2
4 2
2 5
6 5
4 6
```

output

```
4 6 3 1 2 5
```

B. Bridges

2 seconds, 256 megabytes

You are given an undirected graph, not necessarily connected, but with no loops or multiple edges. Find all the bridges in this graph.

Input

The first line of the input file contains two integers n and m ($1 \leq n \leq 20\,000$, $1 \leq m \leq 200\,000$) — the number of vertices and the number of edges of the graph respectively. The next m lines contain a description of edges, with one edge per line. Each edge is described by two integers b_i, e_i ($1 \leq b_i, e_i \leq n$) — the identifiers of connected vertices.

Output

The first line of the output should contain one integer b — the number of the bridges in the graph. In the next line output b integers — the identifiers of edges which are bridges in the increasing order. The edges are numbered from 1 to m in the order of the input.

input

```
6 7
1 2
2 3
3 4
1 3
4 5
4 6
5 6
```

output

```
1
3
```

C. Articulation points

2 seconds, 256 megabytes

Find all articulation points of a given undirected graph.

Input

The first line of the input contains two integers n and m ($1 \leq n \leq 20\,000$, $1 \leq m \leq 200\,000$) — the number of vertices and the number edges of the graph respectively. The next m lines contain the description of edges, with one edge per line. Each edge is described by two integers b_i, e_i ($1 \leq b_i, e_i \leq n$) — the identifiers of connected vertices.

Output

The first line of the output should contain one integer b — the number of articulation points in the graph. The next line should contain b numbers — the identifiers of vertices which are articulation points in the increasing order.

input
<pre> 6 7 1 2 2 3 2 4 2 5 4 5 1 3 3 6 </pre>
output
<pre> 2 2 3 </pre>

D. Edge-biconnected components

2 seconds, 64 megabytes

The edge-biconnected component of a graph $\langle V, E \rangle$ is a subset of vertices $S \subset V$ with a property that for any distinct vertices $u, v \in S$ there exist at least two paths from u to v with no edge in common and there is no set S' such that this property holds for S' and $S \subset S'$.

Given undirected graph. Find all the edge-biconnected components in it.

Input
The first line of the input file contains two integers n and m — number of vertices and edges of the graph respectively ($1 \leq n \leq 20\,000, 1 \leq m \leq 200\,000$). The next m lines contain description of edges, with one edge per line. Edge number i is described by two integers b_i, e_i — the numbers of connected vertices ($1 \leq b_i, e_i \leq n$).

Output
In the first line print one integer k — number of edge-biconnected components of the graph. In the next line print n positive integers a_1, a_2, \dots, a_n , not exceeding k , where a_i — the number of the edge-biconnected component the vertex i belongs to.

input
<pre> 6 7 1 2 2 3 3 1 1 4 4 5 4 6 5 6 </pre>
output
<pre> 2 1 1 1 2 2 2 </pre>

Statement is not available on English language

Е. Компоненты вершинной двусвязности

2 секунды, 64 мегабайта

Компонентой вершинной двусвязности графа $\langle V, E \rangle$ называется максимальный по включению подграф (состоящий из вершин и ребер), такой что любые два ребра из него лежат на вершинно простом цикле.

Дан неориентированный граф без петель. Требуется выделить компоненты вершинной двусвязности в нем.

Входные данные
Первая строка входного файла содержит два натуральных числа n и m — количества вершин и ребер графа соответственно ($1 \leq n \leq 20\,000, 1 \leq m \leq 200\,000$).

Следующие m строк содержат описание ребер по одному на строке. Ребро номер i описывается двумя натуральными числами b_i, e_i — номерами концов ребра ($1 \leq b_i, e_i \leq n$).

Выходные данные
В первой строке выходного файла выведите целое число k — количество компонент вершинной двусвязности графа. Во второй строке выведите m натуральных чисел a_1, a_2, \dots, a_m , не превосходящих k , где a_i — номер компоненты вершинной двусвязности, которой принадлежит i -е ребро. Ребра нумеруются с единицы в том порядке, в котором они заданы во входном файле.

входные данные
<pre> 5 6 1 2 2 3 3 1 1 4 4 5 5 1 </pre>

выходные данные
2 1 1 1 2 2 2

F. Graph condensation

2 seconds, 256 megabytes

Find the number of edges in a condensation of a given oriented graph. Note: condensation doesn't have multiedges.

Input

The first line of the input file contains two integers n and m — the number of vertices and the number of edges respectively ($n \leq 10\,000$, $m \leq 100\,000$). Next m lines contain edges description, one line describes one edge. Edge number i is represented by two numbers b_i e_i — the start and the end of the edge respectively($1 \leq b_i$, $e_i \leq n$). Graph may have multiedges and loops.

Output

Print one integer —the number of edges in a condensation of the graph.

input
4 4 2 1 3 2 2 3 4 3
output
2

Statement is not available on English language

G. Планирование вечеринки

2 секунды, 512 мегабайт

Петя планирует вечеринку, это дело непростое. Одна из главных проблем в том, что некоторые его друзья плохо ладят друг с другом, а некоторые — наоборот. В результате у него есть множество требований, например: «Я приду только если придет Гена» или «Если там будет Марина, то меня там точно не будет».

Петя формализовал все требования в следующем виде: « $[+ -]name1 \Rightarrow [+ -]name2$ », здесь « $name1$ » и « $name2$ » — имена двух друзей Пети, « $+$ » означает, что друг придет в гости, « $-$ » — что не придет. Например, выражение «Если Андрея не будет, то Даша не придет» записывается так: « $-andrey \Rightarrow -dasha$ ».

Помогите Пете составить хоть какой-нибудь список гостей, удовлетворяющий всем свойствам, или скажите, что это невозможно

Входные данные

В первой строке входного файла записаны числа n и m — число друзей Пети и число условий ($1 \leq n$, $m \leq 1000$). В следующих n строках записаны имена друзей. Имена друзей состоят из маленьких латинских букв и имеют длину не больше 10. В следующих m строках записаны условия.

Выходные данные

Выведите в первой строке число k — число друзей, которых нужно пригласить. В следующих k строках выведите их имена.

входные данные
3 3 vova masha gosha -vova => -masha -masha => +gosha +gosha => +vova
выходные данные
2 vova masha

входные данные
1 1 vova -vova => +vova
выходные данные
1 vova

входные данные
2 4 vova masha +vova => +masha +masha => -vova -vova => -masha -masha => +vova
выходные данные
-1

H. Air travel

2 seconds, 256 megabytes

The chief designer Petya was asked to develop a new aircraft model for the company "Air Bubundia". It turned out that the most difficult part is the selection of the optimal size of the fuel tank.

The chief cartographer of "Air Bubundia" Vasya made a detailed map of Bubundia. On this map, he noted the fuel consumption for the flight between each pair of cities.

Petya wants to make the size of the tank as small as possible, such that the plane can fly from any city to any other (possibly with refueling on the way).

Input
First line contains integer n ($1 \leq n \leq 1000$) — number of cities in Bubundia.
Next n lines contain n integers each. j -th number in i -th line equals to the consumption of fuel for the flight from i -th city to j -th city. All number are non-negative and doesn't exceed 10^9 . i -th number in i -th row equals to zero.

Output
Output single integer — optimal tank size.

input
4 0 10 12 16 11 0 8 9 10 13 0 22 13 10 17 0
output
10

Statement is not available on English language

I. Остовное дерево

4 секунды, 256 мегабайт

Даны точки на плоскости, являющиеся вершинами полного графа. Вес ребра равен расстоянию между точками, соответствующими концам этого ребра. Требуется в этом графе найти остовное дерево минимального веса.

Входные данные
Первая строка входного файла содержит натуральное число n — количество вершин графа ($1 \leq n \leq 10\,000$). Каждая из следующих n строк содержит два целых числа x_i, y_i — координаты i -й вершины ($-10\,000 \leq x_i, y_i \leq 10\,000$). Никакие две точки не совпадают.

Выходные данные
Первая строка выходного файла должна содержать одно вещественное число — вес минимального остовного дерева.

входные данные
2 0 0 1 1
выходные данные
1.4142135624

J. Spanning Tree

2 seconds, 256 megabytes

For a given connected undirected graph find a spanning tree with minimum weight.

Input
First line of input consists of two integers n and m — number of vertices and edges, respectively ($2 \leq n \leq 200\,000, 1 \leq m \leq 200\,000$).

Next m lines describe edges one per line in the following format: three integers b_i , e_i and w_i — ends and the weight of the edge i , respectively ($1 \leq b_i, e_i \leq n$, $0 \leq w_i \leq 100\,000$).

"Everything is connected" © D.G (Graph is too, by the way:)

Output

Output a single integer — minimum weight of the spanning tree.

input
4 4 1 2 1 2 3 2 3 4 5 4 1 4
output
7

Statement is not available on English language

К. Минимальное дерево путей

6 секунд, 256 мегабайт

Вам дан взвешенный ориентированный граф, содержащий n вершин и m рёбер. Найдите минимально возможную сумму весов $n - 1$ ребра, которые нужно оставить в графе, чтобы из вершины с номером 1 по этим ребрам можно было добраться до любой другой вершины.

Входные данные

В первой строке даны два целых числа n и m ($1 \leq n \leq 1\,000$, $0 \leq m \leq 10\,000$) — количество вершин и ребер в графе.

В следующих m строках даны ребра графа. Ребро описывается тройкой чисел a_i , b_i и w_i ($1 \leq a_i, b_i \leq n$; $-10^9 \leq w_i \leq 10^9$) — номер вершины, из которой исходит ребро, номер вершины, в которую входит ребро, и вес ребра.

Выходные данные

Если нельзя оставить подмножество ребер так, чтобы из вершины с номером 1 можно было добраться до любой другой, в единственной строке выведите «NO».

Иначе, в первой строке выведите «YES», а во второй строке выведите минимальную возможную сумму весов ребер, которых необходимо оставить.

входные данные
2 1 2 1 10
выходные данные
NO

входные данные
4 5 1 2 2 1 3 3 1 4 3 2 3 2 2 4 2
выходные данные
YES 6