

А. Задание графа

4 с, 1024 МБ

Ориентированный граф задан списком ребер. Требуется вывести его же в виде списка смежности.

Входные данные

В первой строке заданы числа n и m — количество вершин и ребер графа ($1 \leq n, m \leq 100000$). Далее следуют m строк вида $u_i v_i$, обозначающие, что в графе есть ребро из u_i в v_i ($1 \leq u_i, v_i \leq n$).

Выходные данные

Выведите на первой строке количество вершин в графе n , а затем n строк, на i -й из которых через пробел записаны: число k — количество исходящих ребер из i -й вершины, и еще k чисел — номера концов этих ребер в порядке возрастания.

входные данные
3 3 1 2 1 3 2 3
выходные данные
3 2 2 3 1 3 0

входные данные
3 3 1 2 2 3 3 1
выходные данные
3 1 2 1 3 1 1

В. Компоненты связности

4 с, 1024 МБ

Дан **неориентированный граф** из n вершин и m ребер. Требуется покрасить каждую вершину в какой-либо цвет от 1 до n , чтобы цвета вершин совпадали тогда и только тогда, когда они друг из друга достижимы.

Входные данные

В первой строке даны числа n и m ($1 \leq n, m \leq 1000000$). Далее в m строках перечислены попарно различные пары вершин графа u_i, v_i , между которыми есть ребро ($1 \leq u_i, v_i \leq n$).

Выходные данные

Выведите n чисел через пробел — i -е число равно цвету i -й вершины.

входные данные
3 2 1 2 2 3

выходные данные

1 1 1

входные данные

3 1
1 2

выходные данные

1 1 2

С. Предки в дереве

4 с, 1024 МБ

Дано **подвешенное дерево** из n вершин за вершину номер 1 . Требуется ответить на m запросов вида $u v$ — правда ли, что вершина u является предком вершины v в этом дереве?

Входные данные

В первой строке входного файла заданы число n — количество вершин, и число m — количество запросов ($1 \leq n, m \leq 1000000$). Во второй строке дано $n - 1$ число p_i — номера предков всех вершин, начиная со второй ($1 \leq p_i \leq n$).

Далее в m строках перечислены пары чисел $u_i v_i$ — номера вершин, для которых надо ответить на запрос ($1 \leq u_i, v_i \leq n$).

Выходные данные

Выведите m строк, на каждой из которых «1», если в соответствующем запросе u_i является предком v_i , и «0» иначе (без кавычек).

входные данные
3 3 1 2 1 2 1 3 2 3
выходные данные
1 1 1

входные данные

3 3
1 1
1 2
2 1
2 3

выходные данные

1
0
0

Д. Цикл в графе

4 с, 1024 МБ

Дан **ориентированный граф** на n вершинах и m ребрах. Требуется проверить, есть ли в нем цикл, и вывести его, если он есть.

Входные данные

входные данные
3 3
1 2
1 3
2 3
выходные данные
-1

Входные данные
3 3 1 2 2 3 3 1
Выходные данные
0 1 1

входные данные
4 4 1 2 1 3 3 4 2 4
выходные данные
0 1 1 2

входные данные
3 2 2 1 2 2 3
выходные данные
1 0 -1

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

Входные данные

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

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

Выходные данные

Первая строка выходного файла должна содержать одно натуральное число b — количество мостов в заданном графе. На следующей строке выведите b целых чисел — номера рёбер, которые являются мостами, в возрастающем порядке. Рёбра нумеруются с единицы в том порядке, в котором они заданы во входном файле.

входные данные
6 7 1 2 2 3 3 4 1 3 4 5 4 6 5 6
выходные данные
1 3

I. Компоненты реберной двусвязности

6 с, 1024 МБ

Компонентой реберной двусвязности графа $\langle V, E \rangle$ называется подмножество вершин $S \subset V$, такое что для любых различных u и v из этого множества существует не менее двух реберно не пересекающихся путей из u в v .

Дан неориентированный граф. Требуется выделить компоненты реберной 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. Рёбро 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 реберной двусвязности графа. Во второй строке выведите n натуральных чисел a_1, a_2, \dots, a_n , не превосходящих k , где a_i — номер компоненты реберной двусвязности, которой принадлежит i -я вершина.

входные данные
6 7 1 2 2 3 3 1 1 4 4 5 4 6 5 6

выходные данные
2 1 1 1 2 2 2

J. Конденсация графа

4 с, 1024 МБ

Конденсацией графа G называется новый граф H , где каждой компоненте сильной связности в графе G соответствует вершина из графа H . Рёбро vu в графе H есть тогда и только тогда, когда в графе G существует хотя бы одно ребро из соответствующей v компоненте сильной связности, в компоненту, соответствующую u .

Требуется найти количество ребер в конденсации ориентированного графа.

Примечание: конденсация графа не содержит кратных ребер.

Входные данные

Первая строка входного файла содержит два натуральных числа n и m — количество вершин и ребер графа соответственно ($n \leq 10\,000, m \leq 100\,000$). Следующие m строк содержат описание ребер, по одному на строке. Рёбро номер i описывается двумя натуральными числами b_i, e_i — началом и концом ребра соответственно ($1 \leq b_i, e_i \leq n$). В графе могут присутствовать кратные ребра и петли.

Выходные данные

Единственная строка выходного файла должна содержать одно число — количество ребер в конденсации графа.

входные данные
4 4 2 1 3 2 2 3 4 3
выходные данные
2

K. Планирование вечеринок

4 с, 1024 МБ

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: $\ll [+ -] \text{ name1 } \Rightarrow [+ -] \text{ name2 } \gg$, 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: $\ll - \text{ andrey } \Rightarrow - \text{ dasha } \gg$.

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

L. Остовное дерево

4 с, 1024 МБ

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

M. Остовное дерево 2

4 с, 1024 МБ

Требуется найти в связном графе остовное дерево минимального веса.

Входные данные

Первая строка входного файла содержит два натуральных числа n и m — количество вершин и ребер графа соответственно. Следующие m строк содержат описание ребер по одному на строке. Ребро номер i описывается тремя натуральными числами b_i, e_i и w_i — номера концов ребра и его вес соответственно ($1 \leq b_i, e_i \leq n$, $0 \leq w_i \leq 100\,000$). $n \leq 200\,000$, $m \leq 200\,000$.

Граф является связным.

Выходные данные

Первая строка выходного файла должна содержать одно натуральное число — вес минимального остовного дерева.

входные данные
4 4 1 2 1 2 3 2 3 4 5 4 1 4
выходные данные
7

N. Кратчайший путь длины K

4 секунды, 1024 МБ

Дан ориентированный граф. Найдите кратчайшие пути, состоящие из K ребер, от S до всех вершин.

Входные данные

В первой строке дано целых четыре целых числа:

$1 \leq N, M \leq 10^4$ — количества вершин и ребер, $0 \leq K \leq 100$ — количество ребер в кратчайших путях, $1 \leq S \leq N$ — начальная вершина.

В последующих M строках даны тройки целых чисел a_i, b_i, w — начало и конец ребра, а также его вес ($1 \leq a_i, b_i \leq N$, $-10^5 \leq w \leq 10^5$).

Выходные данные

Выведите ровно N чисел по одному в строке. i -е число — длина минимального пути из ровно K ребер из S в i , или -1 , если пути не существует.

входные данные
3 3 1 1 1 2 100 2 3 300 1 3 2
выходные данные
-1 100 2

входные данные
3 3 2 1 1 2 100 2 3 300 1 3 2

выходные данные
-1 -1 400

О. Цикл отрицательного веса

4 с, 1024 МБ

Дан ориентированный граф. Определите, есть ли в нем цикл отрицательного веса, и если да, то выведите его.

Входные данные

Во входном файле в первой строке число N ($1 \leq N \leq 100$) — количество вершин графа. В следующих N строках находится по N чисел — матрица смежности графа. Все веса ребер не превышают по модулю 10 000. Если ребра нет, то соответствующее число равно 100 000.

Выходные данные

В первой строке выходного файла выведите «YES», если цикл существует или «NO» в противном случае. При его наличии выведите во второй строке количество вершин в искомом цикле и в третьей строке — вершины входящие в этот цикл в порядке обхода.

входные данные
2 0 -1 -1 0
выходные данные
YES 2 2 1

Р. Кратчайшие пути

4 с, 1024 МБ

Вам дан взвешенный ориентированный граф и вершина s в нём. Для каждой вершины графа u выведите длину кратчайшего пути от вершины s до вершины u .

Входные данные

Первая строка входного файла содержит три целых числа n , m , s — количество вершин и ребёр в графе и номер начальной вершины соответственно ($2 \leq n \leq 2\,000$, $1 \leq m \leq 5\,000$).

Следующие m строчек описывают рёбра графа. Каждое ребро задаётся тремя числами — начальной вершиной, конечной вершиной и весом ребра соответственно. Вес ребра — целое число, не превосходящее 10^{15} по абсолютной величине. В графе могут быть кратные рёбра и петли.

Выходные данные

Выведите n строчек — для каждой вершины u выведите длину кратчайшего пути из s в u . Если не существует пути между s и u , выведите «*». Если не существует кратчайшего пути между s и u , выведите «-».

входные данные
6 7 1 1 2 10 2 3 5 1 3 100 3 5 7 5 4 10 4 3 -18 6 1 -1
выходные данные
0 10 - - - *

Q. Кратчайший путь

4 с, 1024 МБ

Дан ориентированный взвешенный граф без ребер отрицательного веса. Найдите кратчайшее расстояние от одной заданной вершины до другой.

Входные данные

В первой строке входного файла три числа: n , s и f ($1 \leq n \leq 2000$, $1 \leq s, f \leq n$), где n — количество вершин графа, s — начальная вершина, а f — конечная. В следующих n строках по n чисел — матрица смежности графа, где -1 означает отсутствие ребра между вершинами, а любое неотрицательное число — присутствие ребра данного веса. Вес каждого ребра не превышает 10^9 . На главной диагонали матрицы всегда нули.

Выходные данные

Вывести искомое расстояние или -1, если пути между указанными вершинами не существует.

входные данные
3 1 2 0 -1 2 3 0 -1 -1 4 0
выходные данные
6

Р. Кратчайший путь-2

4 с, 1024 МБ

Дан неориентированный связный взвешенный граф. Найдите кратчайшее расстояние от первой вершины до всех вершин.

Входные данные

В первой строке входного файла два числа: n и m ($2 \leq n \leq 30000$, $1 \leq m \leq 400000$), где n — количество вершин графа, а m — количество ребер.

Следующие m строк содержат описание ребер. Каждое ребро задается стартовой вершиной, конечной вершиной и весом ребра. Вес каждого ребра — неотрицательное целое число, не превосходящее 10^4 .

Выходные данные

Выведите n чисел — для каждой вершины кратчайшее расстояние до нее.

входные данные
<pre> 4 5 1 2 1 1 3 5 2 4 8 3 4 1 2 3 3 </pre>
выходные данные
<pre> 0 1 4 5 </pre>

S. Флойд

4 с, 1024 МБ

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 -ое число в i -ой 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 -ое число в i -ой строке равно весу кратчайшего пути из вершины i в j .

входные данные
<pre> 4 0 5 9 100 100 0 2 8 100 100 0 7 4 100 100 0 </pre>
выходные данные
<pre> 0 5 7 13 12 0 2 8 11 16 0 7 4 9 11 0 </pre>