

Задача А. Компоненты сильный связности

Имя входного файла: `strong.in`
Имя выходного файла: `strong.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Дан ориентированный невзвешенный граф. Необходимо выделить в нем компоненты сильной связности и топологически их отсортировать.

Формат входного файла

В первой строке входного файла находятся два натуральных числа N и M ($1 \leq N \leq 20\,000, 1 \leq M \leq 200\,000$) — количество вершин и рёбер в графе, соответственно. Далее в M строках перечислены рёбра графа. Каждое ребро задаётся парой чисел — номерами начальной и конечной вершин соответственно.

Формат выходного файла

Первая строка выходного файла должна содержать целое число k — количество компонент сильной связности в графе. Вторая строка выходного файла должна содержать n чисел — для каждой вершины выведите номер компоненты сильной связности, которой она принадлежит. Компоненты должны быть занумерованы таким образом, чтобы для каждого ребра (u, v) номер компоненты, которой принадлежит u не превосходил номер компоненты, которой принадлежит v .

Пример

<code>strong.in</code>	<code>strong.out</code>
6 7 1 2 2 3 3 1 4 5 5 6 6 4 2 4	2 1 1 1 2 2 2

Задача В. Конденсация графа

Имя входного файла: `cond.in`
Имя выходного файла: `cond.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Дан ориентированный невзвешенный граф. Необходимо найти его конденсацию.

Формат входного файла

В первой строке входного файла находятся два натуральных числа N и M ($1 \leq N \leq 20\,000, 1 \leq M \leq 200\,000$) — количество вершин и рёбер в графе соответственно. Далее в M строках перечислены рёбра графа. Каждое ребро задаётся парой чисел — номерами начальной и конечной вершин соответственно.

Формат выходного файла

Выведите в выходной файл два числа: число вершин и ребер в конденсации графа.

Пример

cond.in	cond.out
6 7 1 2 2 3 3 1 4 5 5 6 6 4 2 4	2 1

Задача С. Мосты

Имя входного файла: `bridges.in`
Имя выходного файла: `bridges.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайта

Дан неориентированный граф. Требуется найти все мосты в нем.

Формат входного файла

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

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

Формат выходного файла

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

Пример

bridges.in	bridges.out
6 7	1
1 2	3
2 3	
3 4	
1 3	
4 5	
4 6	
5 6	

Задача D. Точки сочленения

Имя входного файла: `points.in`
Имя выходного файла: `points.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайта

Дан неориентированный граф. Требуется найти все точки сочленения в нем.

Формат входного файла

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

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

Формат выходного файла

Первая строка выходного файла должна содержать одно натуральное число b — количество точек сочленения в заданном графе. На следующей строке выведите b целых чисел — номера вершин, которые являются точками сочленения, в возрастающем порядке.

Пример

<code>points.in</code>	<code>points.out</code>
9 12	3
1 2	1 2 3
2 3	
4 5	
2 6	
2 7	
8 9	
1 3	
1 4	
1 5	
6 7	
3 8	
3 9	

Задача Е. Пожарные депо

Имя входного файла: `fire.in`
Имя выходного файла: `fire.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайта

В городе есть n перекрестков, некоторые из которых соединены односторонними дорогами. Всего в городе m дорог.

Мэр решил построить несколько пожарных депо. Для этого ему надо выбрать, на каких перекрестках их построить. При этом когда пожарная машина едет на пожар, она может ехать как по направлению движения по дороге, так и против этого направления. А вот при возвращении в депо срочности нет и ехать «по встречке» уже нельзя.

Помогите мэру построить минимальное число депо, чтобы при пожаре дома на любом перекрестке пожарная машина из одного из депо могла доехать до этого перекрестка (возможно против движения), а затем могла вернуться в свое депо (уже двигаясь только по направлению дорог).

Формат входного файла

В первой строке входного файла находятся два натуральных числа n и m ($1 \leq n \leq 20\,000, 1 \leq m \leq 200\,000$) — количество вершин и улиц. Далее в m строках перечислены улицы, каждая улица задаётся парой чисел — номерами начального и конечного перекрестка.

Формат выходного файла

Выведите одно число — минимальное число депо, которые необходимо построить.

Пример

fire.in	fire.out
6 7 1 2 2 3 3 1 1 4 1 5 4 6 6 4	2

В приведенном примере можно построить депо, например, на перекрестках 4 и 5.

Задача F. Сжигая мосты

Имя входного файла: `burning.in`
Имя выходного файла: `burning.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Остралия состоит из n островов, соединенных m мостами. Все мосты очень симпатичные, все в Остралии очень любят мосты. Конечно же с любого острова можно добраться до любого другого по мостам.

Но недавно агрессивная Захавландия захватила Остралию и король Захавландии приказал сжечь все мосты. К счастью, умный советник подсказал ему, что если он так сделает, то его собственные войска не смогут перемещаться между островами. Поэтому король решил сжечь максимальное число мостов, чтобы по оставшимся мостам все еще можно было добраться с каждого острова до каждого.

Жители Остралии не могу выяснить, какие мосты будут сожжены, но один умный мальчик сказал, что можно вычислить, какие мосты точно не будут сожжены. К сожалению, на следующий день он куда-то пропал.

Помогите тогда вы выяснить, какие мосты точно не будут сожжены королем Захавландии.

Формат входного файла

Первая строка входного файла содержит два натуральных числа n и m — количество островов и мостов, соответственно ($n \leq 20\,000$, $m \leq 200\,000$).

Следующие m строк содержат описание мостов по одному на строке. Мост номер i описывается двумя натуральными числами b_i , e_i — номерами островов, которые он соединяет ($1 \leq b_i, e_i \leq n$). Между парой островов может быть несколько мостов.

Формат выходного файла

Первая строка выходного файла должна содержать одно натуральное число b — количество мостов, которые точно не будут сожжены. На следующей строке выведите b целых чисел — номера мостов, которые точно не будут сожжены. Мосты нумеруются с единицы в том порядке, в котором они заданы во входном файле.

Пример

burning.in	burning.out
6 7	2
1 2	3 7
2 3	
2 4	
5 4	
1 3	
4 5	
3 6	