

Задача А. Кратчайший путь от каждой вершины до каждой

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

Задан ориентированный взвешенный связный граф. Найдите матрицу расстояний между его вершинами.

Формат входного файла

Первая строка входного файла содержит числа n и m — количество вершин и ребер в графе соответственно ($1 \leq n \leq 200$, $0 \leq m \leq 10\,000$). Следующие m строк содержат по три числа — вершины, которые соединяет соответствующее ребро графа и его вес. Веса ребер неотрицательны и не превышают 10^4 .

Формат выходного файла

Выведите в выходной файл n строк по n чисел — для каждой пары вершин выведите расстояние между ними.

Примеры

<code>pathsg.in</code>	<code>pathsg.out</code>
3 3 1 2 5 2 3 2 3 1 8	0 5 7 10 0 2 8 13 0

Задача В. Конденсация графа

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

Вам задан связный ориентированный граф с n вершинами и m ребрами ($1 \leq n \leq 20\,000$, $1 \leq m \leq 200\,000$). Найдите компоненты сильной связности заданного графа и топологически отсортируйте его конденсацию.

Формат входного файла

Граф задан во входном файле следующим образом: первая строка содержит числа n и m . Каждая из следующих m строк содержит описание ребра — два целых числа из диапазона от 1 до n — номера начала и конца ребра, соответственно.

Формат выходного файла

На первой строке выведите число k — количество компонент сильной связности в заданном графе. На следующей строке выведите n чисел — для каждой вершины выведите номер компоненты сильной связности, которой принадлежит эта вершина. Компоненты сильной связности должны быть занумерованы таким образом, чтобы для любого ребра номер компоненты сильной связности его начала не превышал номера компоненты сильной связности его конца.

Примеры

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