

Задача А. Кратчайший путь в невзвешенном графе

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды

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

Формат входных данных

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 и конечной вершиной. Вершины нумеруются с единицы.

Формат выходных данных

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

Пример

стандартный ввод	стандартный вывод
2 1 2 1	0 1

Задача В. Кратчайший путь

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды

Дан ориентированный взвешенный граф. Найдите кратчайшее расстояние от одной заданной вершины до другой.

Формат входных данных

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.

Формат выходных данных

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

Пример

стандартный ввод	стандартный вывод
3 1 2 0 -1 2 3 0 -1 -1 4 0	6

Задача С. Кратчайший путь от каждой вершины до каждой

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды

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

Формат входных данных

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 выходных данных

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

Примеры

стандартный ввод	стандартный вывод
3 3	0 5 7
1 2 5	10 0 2
2 3 2	8 13 0
3 1 8	

Задача D. Кратчайший путь

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды

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

Формат входных данных

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 .

Формат выходных данных

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

Пример

стандартный ввод	стандартный вывод
4 5 1 2 1 1 3 5 2 4 8 3 4 1 2 3 3	0 1 4 5

Задача Е. Кратчайшие пути и прочее

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды

Дан взвешенный ориентированный граф и вершина s в нем. Требуется для каждой вершины u найти длину кратчайшего пути из s в u .

Формат входных данных

Первая строка входного файла содержит n , m и s — количество вершин, ребер и номер выделенной вершины соответственно ($2 \leq n \leq 2000$, $1 \leq m \leq 5000$).

Следующие m строк содержат описание ребер. Каждое ребро задается стартовой вершиной, конечной вершиной и весом ребра. Вес каждого ребра — целое число, не превосходящее 10^{15} по модулю. В графе могут быть кратные ребра и петли.

Формат выходных данных

Выведите n строк — для каждой вершины u выведите длину кратчайшего пути из s в u , '*' если не существует путь из s в u и '-' если не существует кратчайший путь из s в u .

Пример

стандартный ввод	стандартный вывод
6 7 1	0
1 2 10	10
2 3 5	-
1 3 100	-
3 5 7	-
5 4 10	*
4 3 -18	
6 1 -1	

Задача F. Цикл отрицательного веса

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды

Дан ориентированный взвешенный граф. Определить, есть ли в нем цикл отрицательного веса, и если да, то вывести его.

Формат входных данных

Во входном файле в первой строке число n ($1 \leq n \leq 250$) — количество вершин графа. В следующих n строках находится по n чисел — матрица смежности графа. Все веса ребер не превышают по модулю 10000. Если ребра нет, то соответствующее число равно 10^9 .

Формат выходных данных

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 (считая одинаковые первую и последнюю) и в третьей строке — вершины, входящие в этот цикл в порядке обхода.

Примеры

стандартный ввод	стандартный вывод
2	YES
0 -1	3
-1 0	1 2 1

Задача G. Остовное дерево

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта

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.

Формат входных данных

Первая строка входного файла содержит натуральное число n — количество вершин графа ($1 \leq n \leq 5000$). 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.

Формат выходных данных

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

Примеры

стандартный ввод	стандартный вывод
3 0 0 1 0 0 1	2

Задача Н. Остовное дерево 2

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Требуется найти в связном графе остовное дерево минимального веса.

Формат входных данных

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 ($1 \leq b_i, e_i \leq n$, $0 \leq w_i \leq 100\,000$). $n \leq 20\,000$, $m \leq 100\,000$.

Граф является связным.

Формат выходных данных

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

Примеры

стандартный ввод	стандартный вывод
4 4 1 2 1 2 3 2 3 4 5 4 1 4	7

Задача I. Плотное остовное дерево

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 4 секунды
Ограничение по памяти: 64 мегабайта

Требуется найти в графе остовное дерево, в котором разница между весом максимального и минимального ребра минимальна.

Формат входных данных

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 ($1 \leq b_i, e_i \leq n$, $0 \leq |w_i| \leq 10^9$). $n \leq 1000$, $m \leq 10\,000$.

Формат выходных данных

Если остовное дерево существует, выведите в первой строке выходного файла YES, а во второй строке одно целое число — минимальную разность между весом максимального и минимального ребра в остовном дереве.

В противном случае в единственной строке выведите NO.

Примеры

стандартный ввод	стандартный вывод
4 5 1 2 1 1 3 2 1 4 1 3 2 2 3 4 2	YES 0

Задача J. Алгоритм двух китайцев

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 4 секунды

Дан ориентированный взвешенный граф. Покрывающим деревом с корнем в вершине u назовем множество ребер, таких что из вершины u достижима любая другая вершина v , притом единственным образом. Весом дерева назовем сумму весов его ребер.

Требуется определить, существует ли в данном графе покрывающее дерево с корнем в вершине с номером 1. В случае существования требуется определить его минимальный вес.

Формат входных данных

В первой строке входного файла два числа: n и m ($2 \leq n \leq 1000, 1 \leq m \leq 10000$), 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 — целое число, не превосходящее по модулю 10^9 .

Формат выходных данных

Если покрывающее дерево существует, выведите в первой строке выходного файла YES, а во второй строке целое число — его минимальный вес. В противном случае в единственной строке выведите NO.

Примеры

стандартный ввод	стандартный вывод
2 1 2 1 10	NO
4 5 1 2 2 1 3 3 1 4 3 2 3 2 2 4 2	YES 6

Задача К. Алгоритм двух китайцев — 2

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 4 секунды
Ограничение по памяти: 512 мегабайт

Вам дан взвешенный ориентированный граф. Найдите минимальную возможную сумму весов ребер, которых необходимо оставить в графе, чтобы из вершины с номером 1 по этим ребрам можно было добраться до любой другой вершины.

Формат входных данных

В первой строке даны два целых числа n и m ($1 \leq n \leq 300\,000$, $0 \leq m \leq 300\,000$) — 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.

Формат выходных данных

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.

Примеры

стандартный ввод	стандартный вывод
2 1 2 1 10	NO
4 5 1 2 2 1 3 3 1 4 3 2 3 2 2 4 2	YES 6