

Лабораторная работа 2-2. Графы, кратчайшие пути

А. Флойд

2 секунды, 256 мегабайт

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
4 0 5 9 100 100 0 2 8 100 100 0 7 4 100 100 0
выходные данные
0 5 7 13 12 0 2 8 11 16 0 7 4 9 11 0

В. Кратчайший путь-2

2 секунды, 256 мегабайт

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
4 5 1 2 1 1 3 5 2 4 8 3 4 1 2 3 3
выходные данные
0 1 4 5

С. Цикл отрицательного веса

2 секунды, 256 мегабайт

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

Кратчайший путь длины  $K$

4 секунды, 256 мегабайт

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.

d. кратчайший путь длины $k$
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

Е. Кратчайшие пути

2 секунды, 256 мегабайт

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 - - - *

Г. В поисках утраченного кефира

2 секунды, 256 мегабайт

Школьник Вася хочет найти запасы спрятанного кефира. По легенде, кефир находится в домиках  $a, b$  или  $c$ . Вася хочет проверить каждый из этих трёх домиков, потратив на это минимальное количество времени.

Местность, в которой находится Вася представляет собой  $n$  домиков, пронумерованных числами от 1 до  $n$ . Некоторые из домиков соединены дорогами, по которым можно ходить в обе стороны. Время прохождения  $i$ -й дороги составляет  $w_i$  секунд. Путём в графе называется непустая последовательность вершин, такая что все соседние вершины соединены дорогой. Требуется помочь Васе найти путь, содержащий вершины  $a, b, c$ , такой что суммарное время прохождения всех дорог на пути минимально. При этом, если мы прошли по какой-то дороге дважды (или более), то и время её прохождения следует учитывать соответствующее количество раз. Начинать свой путь Вася может из любой вершины.

Гарантируется, что  $a, b, c$  — попарно различные домики.

Входные данные

В первой строке ввода записаны два числа  $n$  и  $m$  ( $3 \leq n \leq 100\,000, 0 \leq m \leq 200\,000$ ) — количество домиков в ЛКШ и дорог между ними соответственно.

Следующие  $m$  строк содержат описания дорог, по одному в строке. Каждая из дорог задаётся тройкой чисел  $u_i, v_i, w_i$  ( $1 \leq u_i, v_i \leq n, 1 \leq w_i \leq 10^9$ ) — номерами соединённых домиков и временем, затрачиваемым на прохождение данной дороги. По каждой дороге разрешено ходить в обе стороны. Гарантируется, что любая пара домиков соединена не более чем одной дорогой. Также гарантируется, что нет дороги, соединяющей домик с самим собой.

В последней строке записаны три попарно различных числа  $a, b, c$  ( $1 \leq a, b, c \leq n$ ).

Выходные данные

Выведите одно целое число — минимальное возможное время, которое нужно затратить на прохождение пути, содержащего домики  $a, b$  и  $c$ . Если пути, содержащего все три домика не существует, то выведите -1.

входные данные
4 4 1 2 3 2 3 1 3 4 7 4 2 10 1 4 3
выходные данные
11

входные данные
4 2 1 2 10 2 3 5 1 2 4
выходные данные
-1

В первом примере путь 1–2–3–4 является минимальным (11 секунд). Например, путь 1–2–4–3 не подходит, так как занимает больше времени (20 секунд), а путь 3–4–2 не подходит, так как домик  $a$  оказывается не посещенным.

Во втором примере не существует способа добраться от домика  $b$  до домика  $c$ , поэтому искомого пути не существует.

Г. Бемби

2 секунды, 256 мегабайт

Существует страна, в которой  $n$  городов. Города пронумерованы от 1 до  $n$ . Также в этой стране существуют двунаправленные дороги. Каждая дорога соединяет пару городов. Для каждого  $i$ , автомобильная дорога  $i$  соединяет города  $a_i$  и  $b_i$ .

Бемби — это олень, который любит путешествовать по дорогам. Движение по дороге  $i$  (в любом направлении) занимает у оленя  $d_i$  минут. Бемби ненавидит города и из-за этого никогда в них не задерживается.

Бемби начинает путешествие из города номер 1. Через  $t$  минут он желает оказаться в городе  $n$ . Вы должны узнать, может ли Бемби достигнуть город  $n$  ровно через  $t$  минут.

Входные данные

Первая строка содержит два целых числа  $n$  и  $m$  — количество городов и дорог в стране ( $1 \leq n, m \leq 50$ ).

Следующие  $m$  строк описывают дороги. Каждая строка состоит из чисел  $a_i, b_i$  и  $d_i$  — концы дороги и ее длина ( $1 \leq a_i, b_i \leq n, 1 \leq d_i \leq 10^4$ ).

Последняя строка содержит целое число  $t$  — количество минут, за которое Бемби желает добраться до города  $n$  ( $1 \leq t \leq 10^{18}$ ).

Выходные данные

Выведите "Possible" если Бемби сможет достичь цели ровно за  $t$  минут, иначе выведите "Impossible".

входные данные
3 3 1 3 7 1 2 6 2 3 5 11
выходные данные
Possible

<b>входные данные</b>
4 3 1 3 10 1 2 10 2 3 10 1000
<b>выходные данные</b>
Impossible

## H. Dwarf Tower

3 seconds, 256 megabytes

Little Vasya is playing a new game named "Dwarf Tower". In this game there are  $n$  different items, which you can put on your dwarf character. Items are numbered from 1 to  $n$ . Vasya wants to get the item with number 1.

There are two ways to obtain an item:

- You can buy an item. The  $i$ -th item costs  $c_i$  money.
- You can craft an item. This game supports only  $m$  types of crafting. To craft an item, you give two particular different items and get another one as a result.

Help Vasya to spend the least amount of money to get the item number 1.

The first line of input contains two integers  $n$  and  $m$  ( $1 \leq n \leq 200\,000; 0 \leq m \leq 500\,000$ ) — the number of different items and the number of crafting types.

The second line contains  $n$  integers  $c_i$  — values of the items ( $0 \leq c_i \leq 10^9$ ).

The following  $m$  lines describe crafting types, each line contains three distinct integers  $a_i, x_i, y_i$  —  $a_i$  is the item that can be crafted from items  $x_i$  and  $y_i$  ( $1 \leq a_i, x_i, y_i \leq n$ ;  $a_i \neq x_i$ ;  $x_i \neq y_i$ ;  $y_i \neq a_i$ ).

The output should contain a single integer — the least amount of money to spend.

input
5 3 5 0 1 2 5 5 2 3 4 2 3 1 4 5
output
2