

Содержание

Must have	2
Задача 9A. Сумма расстояний [0.3 (1.5), 256 mb]	2
Задача 9B. Расстояние между вершинами [0.5 sec (5.0), 256 mb]	3
Задачи здорового человека	4
Задача 9C. Стоимость проезда [0.2 sec (2.5), 256 mb]	4
Задача 9D. Longpath. Длиннейший путь [0.2 sec (2.0), 256 mb]	5
Задача 9E. Расстояние между вершинами [0.3 sec (3.0), 256 mb]	6
Для искателей острых ощущений	7
Задача 9F. Кратчайший путь двух коней [0.2 sec (2.0), 256 mb]	7
Задача 9G. Path. Кратчайший путь [0.5 sec (6.0), 256 mb]	8
Задача 9H. Путь в случайном графе [0.3 sec (2.0), 256 mb]	9

У вас не получается читать/выводить данные?

Воспользуйтесь примерами (c++) (python).

Обратите внимание, входные данные лежат в **стандартном потоке ввода** (он же stdin), вывести ответ нужно в **стандартный поток вывода** (он же stdout).

Must have

Задача 9А. Сумма расстояний [0.3 (1.5), 256 mb]

Дан связный неориентированный граф. Требуется найти сумму расстояний между всеми парами вершин.

Формат входных данных

Первая строка входного файла содержит два натуральных числа n и m — количество вершин и ребер графа соответственно ($1 \leq n \leq 1000$, $0 \leq m \leq 10\,000$).

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

Гарантируется, что граф связан.

Формат выходных данных

Первая строка выходного файла должна содержать одно натуральное число — сумму попарных расстояний между вершинами.

Пример

stdin	stdout
5 5 1 2 2 3 3 4 5 3 1 5	16

Подсказка по решению

Задача *не* про Дейкстру.

Задача 9В. Расстояние между вершинами [0.5 sec (5.0), 256 mb]

Дан взвешенный неориентированный граф. Требуется найти вес минимального пути между двумя вершинами.

Формат входных данных

Первая строка входного файла содержит два числа n и m — количество вершин и ребер графа соответственно. Вторая строка входного файла содержит натуральные числа s и t — номера вершин, длину пути между которыми требуется найти ($1 \leq s, t \leq n, s \neq t$).

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

$1 \leq n \leq 100\,000, 1 \leq m \leq 200\,000$.

Формат выходных данных

Первая строка выходного файла должна содержать одно натуральное число — вес минимального пути между вершинами s и t .

Если путь из s в t не существует, выведите -1 .

Пример

stdin	stdout
4 4 1 3 1 2 1 3 4 5 3 2 2 4 1 4	3

Подсказка по решению

Дейкстра обыкновенная. `set<pair<int,int>>` зайдет. `priority_queue` быстрее.

Эстеты вместо пар могут создать `struct Vertex {int id;}` и перегрузить `operator<`.

Задачи здорового человека

Задача 9С. Стоимость проезда [0.2 sec (2.5), 256 mb]

Страна состоит из n городов и m дорог. Города пронумерованы числами от 1 до n . Город с номером s является столицей. Все дороги односторонние, проход по каждой дороге стоит ровно 1 золотой. Требуется найти минимальные стоимости проезда от каждого города до столицы.

Формат входных данных

В первой строке файла записаны три целых числа — n , s и m (количество городов, номер столичного города и количество дорог).

В следующих m строках записаны пары чисел. Пара чисел (a, b) означает, что есть дорога из города a в город b .

Ограничения: $1 \leq n \leq 10^5$, $0 \leq m \leq 10^5$.

Формат выходных данных

Выведите n чисел — минимальные стоимости проезда от городов до столицы. Если от какого-то города не существует ни одного пути до столицы, выведите -1 .

Пример

stdin	stdout
3 2 2	1 0 -1
1 2	
2 3	

Подсказка по решению

Казалось бы bfs. Или нет? Или всё-таки да?

Задача 9D. Longpath. Длиннейший путь [0.2 sec (2.0), 256 mb]

Дан ориентированный граф без циклов. Требуется найти в нем длиннейший путь.

Формат входных данных

Первая строка входного файла содержит два натуральных числа n и m — количество вершин и дуг графа соответственно. Следующие m строк содержат описания дуг по одной на строке. Ребро номер i описывается двумя натуральными числами b_i и e_i — началом и концом дуги соответственно ($1 \leq b_i, e_i \leq n$).

Входной граф не содержит циклов и петель.

$n \leq 10\,000$, $m \leq 100\,000$.

Формат выходных данных

Первая строка выходного файла должна содержать одно натуральное число — количество дуг в длиннейшем пути.

Пример

stdin	stdout
5 5 1 2 2 3 3 4 3 5 1 5	3

Подсказка по решению

В орграфе без циклов мы умеем почти что угодно считать dfs-ом (динамикой).

Задача 9Е. Расстояние между вершинами [0.3 sec (3.0), 256 mb]

Дан неориентированный взвешенный граф без петель и кратных рёбер. Найти вес минимального пути между двумя вершинами.

Формат входных данных

Первая строка входного файла содержит натуральные числа N , M , вторая строка содержит натуральные числа S и F ($N \leq 5\,000$, $M \leq 100\,000$, $1 \leq S, F \leq N$, $S \neq F$) — количество вершин и ребер графа а также номера вершин, длину пути между которыми требуется найти.

Следующие M строк по три натуральных числа b_i , e_i и w_i — номера концов i -ого ребра и его вес соответственно ($1 \leq b_i, e_i \leq n$, $0 \leq w_i \leq 100\,000$).

Формат выходных данных

Первая строка должна содержать одно натуральное число — вес минимального пути между вершинами S и F . Во второй строке через пробел выведите вершины на кратчайшем пути из S в F в порядке обхода.

Если путь из S в F не существует, выведите -1 .

Пример

stdin	stdout
4 4	3
1 3	1 2 3
1 2 1	
3 4 5	
3 2 2	
4 1 4	

Подсказка по решению

Существует жизнь без кучи.

Матрица смежности в этой задаче *крайне* не эффективна.

Для искателей острых ощущений

Задача 9F. Кратчайший путь двух коней [0.2 sec (2.0), 256 mb]

Переведите каждого из двух коней из одной клетки в другую за наименьшее общее число ходов. Два коня не могут одновременно находиться в одной клетке.

Формат входных данных

Во входном файле записаны координаты первого и второго коня, затем координаты клеток, куда нужно их переместить.

Формат выходных данных

Программа должна вывести последовательность ходов коней в виде нескольких строк. Первым символом в строке должен быть номер коня (1 или 2), затем, через пробел, координаты клетки, в которую он переставляется. Необходимо вывести любое из возможных оптимальных решений.

Пример

stdin	stdout
a1	1 b3
c2	1 d4
c2	2 a1
a1	1 c2

Подсказка по решению

Идейно простая. А вот с кодом нужно повозиться.

Задача 9G. Path. Кратчайший путь [0.5 sec (6.0), 256 mb]

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 ($2 \leq n \leq 2000$, $1 \leq m \leq 6000$).

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

stdin	stdout
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	

Подсказка по решению

Форд-Беллман. Аккуратнее с переполнениями целочисленного типа.

В данной задаче бывает не только overflow, но и underflow.

Поскольку нам нужно найти не какие-то, а все вершины, *достижимые* из отрицательных циклов, недостаточно *сделать n итераций и отметить релаксированные вершины*, нужно делать $2n$ итераций, и помечать вершины, посещённые на $\geq n$ -й.

Задача 9Н. Путь в случайном графе [0.3 sec (2.0), 256 mb]

Дан случайный взвешенный ориентированный граф из n вершин, m рёбер. Гарантируется, что в графе нет отрицательных циклов. Ваша задача – найти расстояние от вершины s до вершины t . Что такое случайный граф? Рёбра равномерно случайного веса между равномерно случайными вершинами.

Формат входных данных

На первой строке $n m s t$ ($1 \leq n \leq 100\,000$, $0 \leq m \leq 200\,000$, $1 \leq s, t \leq n$). На следующих m строках тройки чисел $a_i b_i w_i$ ($1 \leq a, b \leq n$, $|w| < 10^9$), рёбра из a_i в b_i веса w_i .

Формат выходных данных

Одно число – длину кратчайшего пути.

Если пути из s в t не существует, выведите число 10^{18} .

Примеры

stdin	stdout
2 0 1 2	10000000000000000000
3 4 1 3 1 2 20 2 3 -15 1 3 10 3 1 -4	5

Подсказка по решению

Форд-Беллман обычно гораздо быстрее VE .

Задача о том, как писать Форд-Беллмана, чтобы работал в среднем по тестам быстро.