

## Задача А. Компоненты связности

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

Вам задан неориентированный граф с  $N$  вершинами и  $M$  ребрами ( $1 \leq N \leq 20\,000$ ,  $1 \leq M \leq 200\,000$ ). В графе отсутствуют петли и кратные ребра.

Определите компоненты связности заданного графа.

### Формат входных данных

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 концов ребра.

### Формат выходных данных

На первой строке выходного файла выведите число  $L$  — количество компонент связности заданного графа. На следующей строке выведите  $N$  чисел из диапазона от 1 до  $L$  — номера компонент связности, которым принадлежат соответствующие вершины. Компоненты связности следует занумеровать от 1 до  $L$  произвольным образом.

### Пример

connect.in	connect.out
4 2	2
1 2	1 1 2 2
3 4	

## Задача В. Стоимость проезда

Имя входного файла: bfsrev.in  
Имя выходного файла: bfsrev.out  
Ограничение по времени: 0.5 секунды  
Ограничение по памяти: 256 мегабайт

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

bfsrev.in	bfsrev.out
3 2 2	1 0 -1
1 2	
2 3	

## Задача С. Авиаперелеты

Имя входного файла: avia.in  
Имя выходного файла: avia.out  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Профессору Форду необходимо попасть на международную конференцию. Он хочет потратить на дорогу наименьшее количество денег, поэтому решил, что будет путешествовать исключительно ночными авиарейсами (чтобы не тратиться на ночевку в отелях), а днем будет осматривать достопримечательности тех городов, через которые он будет проезжать транзитом. Он внимательно изучил расписание авиаперелетов и выбрал подходящие ему авиарейсы, выяснив, что перелеты на выбранных направлениях совершаются каждую ночь и за одну ночь он не сможет совершить два перелета.

Теперь профессор хочет найти путь наименьшей стоимости, учитывая что до конференции осталось  $K$  ночей (то есть профессор может совершить не более  $K$  перелетов).

### Формат входных данных

В первой строке находятся числа  $N$  (количество городов),  $M$  (количество авиарейсов),  $K$  (количество оставшихся ночей),  $S$  (номер города, в котором живет профессор),  $F$  (номер города, в котором проводится конференция). Ограничения:  $2 \leq N \leq 100$ ,  $1 \leq M \leq 10^5$ ,  $1 \leq K \leq 100$ ,  $1 \leq S \leq N$ ,  $1 \leq F \leq N$ .

Далее идет  $M$  строк, задающих расписание авиарейсов.  $i$ -я строка содержит три натуральных числа:  $S_i$ ,  $F_i$  и  $P_i$ , где  $S_i$  — номер города, из которого вылетает  $i$ -й рейс,  $F_i$  — номер города, в который прилетает  $i$ -й рейс,  $P_i$  — стоимость перелета  $i$ -м рейсом.  $1 \leq S_i \leq N$ ,  $1 \leq F_i \leq N$ ,  $0 \leq P_i \leq 10^6$ .

### Формат выходных данных

Выведите одно число — минимальную стоимость пути, подходящего для профессора. Если профессор не сможет за  $K$  ночей добраться до конференции, выведите

число -1.

### Пример

avia.in	avia.out
4 5 2 1 4 1 2 1 2 3 1 3 4 1 1 3 3 1 4 5	4

## Задача D. Диаметр графа

Имя входного файла: `diameter.in`  
Имя выходного файла: `diameter.out`  
Ограничение по времени: 1 секунды  
Ограничение по памяти: 256 мегабайт

Дан связный взвешенный неориентированный граф.

Рассмотрим пару вершин, расстояние между которыми максимально среди всех пар вершин. Расстояние между ними называется *диаметром графа*. *Эксцентриситетом вершины*  $v$  называется максимальное расстояние от вершины  $v$  до других вершин графа. *Радиусом графа* называется наименьший из эксцентриситетов вершин. Найдите диаметр и радиус графа.

### Формат входных данных

В первой строке входного файла единственное число:  $N$  ( $1 \leq N \leq 100$ ) — количество вершин графа. 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. На главной диагонали матрицы всегда нули; веса рёбер не превышают 1000.

### Формат выходных данных

В выходной файл выведите два числа - диаметр и радиус графа.

### Пример

diameter.in	diameter.out
4 0 -1 1 2 -1 0 -1 5 1 -1 0 4 2 5 4 0	8 5

## Задача E. Расстояние между вершинами

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

Коль Дейкстру́ писать без кучи,  
То тайм-лимит ты получишь...  
А в совсем крутой задаче  
Юзай кучу Фибоначчи!

Спектакль преподавателей ЛКШ.июль–2007

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

### Формат входных данных

Первая строка входного файла содержит два натуральных 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 ( $1 \leq b_i, e_i \leq n$ ,  $0 \leq w_i \leq 100$ ).

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

### 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

distance.in	distance.out
4 4 1 3 1 2 1 3 4 5 3 2 2 4 1 4	3

## Задача F. Опасный маршрут

Имя входного файла: `danger.in`  
Имя выходного файла: `danger.out`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Профессор Дейкстра живёт в очень опасном районе города. Ежедневно бандиты грабят на улицах прохожих. Читая криминальную хронику, профессор вычислил вероятность быть ограбленным при проходе по каждой улице города.

Теперь он хочет найти наиболее безопасный путь от дома до университета, в котором он преподаёт. Иными словами, он хочет найти путь от дома до университета, для которого вероятность быть ограбленным минимальна.

### Формат входных данных

В первой строке входного файла записаны два числа  $N$  и  $M$  — количество зданий и количество улиц, соединяющих здания ( $1 \leq N \leq 100$ ,  $1 \leq M \leq N(N-1)/2$ ). В следующей строке находятся число  $S$  и  $F$  — номер дома, в котором живёт профессор и номер дома, в котором находится университет соответственно. Далее в  $M$  строках расположены описания дорог: 3 целых числа  $S_i$ ,  $F_i$  и  $P_i$  — номера зданий, в которых начинается и заканчивается дорога, и вероятность в процентах быть ограбленным, пройдя по дороге соответственно ( $1 \leq S_i \leq N$ ,  $1 \leq F_i \leq N$ ,  $0 \leq P_i \leq 100$ , дороги двунаправленные). Гарантируется, что существует хотя бы один путь от дома профессора до университета.

### Формат выходных данных

Необходимо вывести одно число — минимальную возможную вероятность быть ограбленным с точностью не менее 6 знаков после запятой.

### Пример

<code>danger.in</code>	<code>danger.out</code>
3 3 1 3 1 2 20 1 3 50 2 3 20	0.36

Пояснение. Для приведенного примера минимальная вероятность быть ограбленным достигается на маршруте  $1 - 2 - 3$ .

Указание. Пусть вероятность быть ограбленным на пути  $A - B$  равна  $p$ , а на пути от  $B - C$  равна  $q$ . Какова вероятность быть ограбленным на пути  $A - B - C$ ?

## Задача G. Наименьшее кратное

Имя входного файла: `multiple.in`  
Имя выходного файла: `multiple.out`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Дано число  $X$  и множество цифр  $D$ . Требуется дописать к  $X$  минимальное количество цифр из  $D$ , чтобы получившееся число делилось на  $k$ . При этом получившееся число должно быть минимально возможным.

### Формат входных данных

Первая строка входного файла содержит два натуральных числа  $X$  и  $k$  ( $1 \leq X \leq 10^{1000}$ ,  $2 \leq k \leq 100\,000$ ). Во второй строке записано количество цифр во множестве  $D$ . В третьей строке через пробел записаны эти цифры.

### Формат выходных данных

Единственная строка должна содержать минимальное число, полученное из  $X$  дописыванием цифр из  $D$  и кратное  $k$ . Если такого числа не существует, выведите -1.

### Пример

<code>multiple.in</code>	<code>multiple.out</code>
102 101 3 1 0 3	10201
202 101 3 1 0 3	202