

Задача 1. Шлю я за пакетом пакет

Источник: базовая
Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

В базе данных роутера хранится информация о нескольких серверах, некоторые из них связаны между собой напрямую, другие — только опосредованно. Получая электронное письмо от сервера с номером M (отправителя), предназначенное для сервера с номером N (получателя), роутер должен найти в своей базе данных самый короткий путь пересылки этого письма по сети.

Напишите программу, которая, зная информацию о всевозможных соединениях и их длительности, осуществляла бы поиск самого короткого пути пересылки и выдавала бы информацию о времени, за которое письмо преодолит весь этот путь.

Формат входных данных

В первой строке входного файла содержится одно натуральное число N — количество серверов, информация о которых записана в базе данных роутера ($1 \leq N \leq 100$).

Во второй строке — два целых числа S_1 и S_2 , разделенные пробелом — номера сервера-отправителя и сервера-получателя ($1 \leq S_1, S_2 \leq N$).

Начиная с третьей строки и до конца файла, записаны имеющиеся между серверами активные каналы связи и скорость передачи данных по этим каналам. Сначала записаны номера серверов S_i и S_j , а затем скорость передачи данных между ними — целое число K ($1 \leq S_i, S_j \leq N, 0 \leq K \leq 1000$). Все числа находятся на одной строке и разделены пробелами.

Формат выходных данных

В выходной файл нужно записать одно целое число — время, необходимое письму для прохождения по самому быстрому пути связи.

Если такого пути нет (например, на промежуточном сервере произошла авария), то необходимо выдать сообщение `no`.

Примеры

<code>input.txt</code>	<code>output.txt</code>
4 4 1 1 2 10 1 3 2 2 4 1 3 4 10	11
4 1 4 1 2 100 1 3 20 2 3 57	no

Задача 2. Фиолетовое такси

Источник: основная
Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

«Фиолетовое такси» предложило клиентам новую услугу — теперь можно узнать кратчайшее время проезда для любых мест отправления и назначения. Однако диспетчеры не успевают отвечать на все запросы.

Вас просят написать программу, которая поможет диспетчерам быстро получать время пути между двумя пунктами.

Формат входных данных

В первой строке входного файла записаны два числа: N — количество мест, на которых люди садятся и выходят из такси, и M — количество дорог, их соединяющих ($1 \leq N \leq 300, 1 \leq M \leq 30\,000$).

Далее в M строках описываются дороги, по три числа в каждой строке — номера двух пунктов, соединенных этой дорогой и время t_i пути в минутах ($1 \leq t_i \leq 10\,000$). Все дороги — двусторонние.

Далее в одной строке записано число K — количество запросов ($1 \leq K \leq 100\,000$).

В следующих K строках даны запросы — номера пункта отправления и пункта назначения. Любую пару мест может соединять несколько дорог.

Формат выходных данных

Для каждого запроса нужно на отдельной строке вывести одно число — время пути в минутах. Гарантируется, что все места соединены между собой.

Пример

input.txt	output.txt
5 6	9
4 2 2	8
1 4 8	
2 3 6	
1 5 7	
2 1 3	
4 3 9	
2	
1 3	
4 3	

Задача 3. Транзитивное замыкание

Источник: основная
Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 1 секунда
Ограничение по памяти: разумное

Дано бинарное отношение R над множеством чисел $X = \{1, 2, 3, \dots, N\}$. Требуется найти его рефлексивно-транзитивное замыкание.

Указание: Используйте алгоритм Уоршалла (Флойда-Уоршалла).

Формат входных данных

В первой строке входного файла записано одно целое число N — размер множества ($1 \leq N \leq 500$).

Далее идёт N строк по N символов в каждой, задающие отношение R . j -ый символ i -ой строки равен 1, если пара $(i, j) \in R$ (т.е. лежит в отношении R), и равен 0 в противном случае.

Формат выходных данных

В выходной файл необходимо вывести N строк по N символов в каждой — рефлексивное и транзитивное замыкание отношения R , описанное в том же формате, что и исходное отношение R во входных данных.

Пример

input.txt	output.txt
5	10011
00001	11011
10010	11111
01101	10011
10000	10011
00011	

Задача 4. Сплетницы

Источник: основная
Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Домохозяйки проводят много времени за телефонными разговорами с подругами. Как только одной из них становится что-нибудь известно, она садится за телефон и сообщает свежую сплетню подругам. С каждой из своих подруг домохозяйка заканчивает разговор через разное время.

Напишите программу, вычисляющую минимальное время, за которое сплетня дойдет от домохозяйки-распространительницы свежей сплетни, живущей в доме K , до другой, живущей в доме L .

Формат входных данных

В первой строке файла содержатся целые числа N , M — количество домохозяек и количество пар любящих поговорить домохозяек ($1 \leq N \leq 50\,000$, $1 \leq M \leq 300\,000$).

Во второй строке записаны целые числа K и L ($1 \leq K, L \leq 50\,000$).

В каждой из следующих строк располагаются по три целых числа, из которых первые два числа — это номера домов домохозяек, которые любят разговаривать по телефону друг с другом, третье число — время разговора между ними — неотрицательно и не превосходит 10 000.

Формат выходных данных

В выходной файл необходимо выдать целое число — минимальное время телефонных разговоров, за которое сплетня из дома K дойдет в дом L . Если же сплетня не может дойти до адресата, то выведите слово NO.

Пример

input.txt	output.txt
7 6 1 7 1 2 2 1 3 1 3 6 1 3 7 2 6 7 1 2 4 1	3

Задача 5. Авиаперелёты

Источник: основная
Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Текст данной задачи был написан более 15 лет назад, когда интернет тарифицировался по мегабайтам, никаких `one-two-trip` и `aviasales` не было, а для покупки билетов, действительно, надо было ножками ходить в турагентства.

Группа туристов решила совершить путешествие из точки A в точку B на самолете. Однако они выяснили, что прямой рейс — не всегда наилучший вариант. Получив прайс-лист всех доступных авиарейсов, туристы начали составлять план самого дешевого маршрута. Однако отпуск туристов закончится раньше, чем они закончат подсчеты вручную. От лица авиакомпании вам нужно предложить программное решение проблемы.

Формат входных данных

В первой строке входного файла записаны два числа: N — количество аэропортов и M — количество авиарейсов ($1 \leq N \leq 100000, 1 \leq M \leq 200000$).

Далее в M строках описываются авиарейсы, по три числа в каждой строке — номер аэропорта отправления, номер аэропорта прибытия и c_i — стоимость перелета ($1 \leq c_i \leq 10000$).

Формат выходных данных

В выходной файл необходимо вывести одно целое число — стоимость самого дешевого маршрута из аэропорта с номером 1 в аэропорт с номером N .

Пример

input.txt	output.txt
5 9 5 3 9 2 1 8 5 4 2 3 4 5 1 2 1 4 5 6 1 3 5 3 5 8 1 5 23	13

Задача 6. Спички

Источник: Повышенной сложности
Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Во время игры какое-то количество спичек рассыпается по столу и игроки должны убрать их, выбирая по одной, не сдвинув с места остальные спички.

Вам нужно определить, касаются ли данные спички друг друга. Вам будет дан список координат концов спичек, и нужно будет определить, соединены ли две данные спички или нет. Заметим, что касание – это соединение, и что две спички могут быть соединены, если они соединены не прямо, а посредством других спичек.

Формат входных данных

Входной файл содержит несколько тестов.

В первой строке каждого теста находится целое число n , задающее количество спичек на столе ($1 < n < 13$).

Каждая из следующих n строк содержит 4 положительных целых x_1, y_1, x_2, y_2 , обозначающих координаты $(x_1, y_1), (x_2, y_2)$ концов одной спички. Все координаты меньше 100. Заметим, что спички могут быть разной длины. Первая введенная спичка будет спичкой номер 1, вторая, соответственно, номер 2 и т.д. Все данные корректны, и нет спичек нулевой длины.

Остальные строки теста содержат по два целых числа a и b (от 1 до n , включительно). Вам нужно определить, связана ли спичка a со спичкой b , или нет.

Конец теста: $a = b = 0$. После каждого теста во входном файле пустая строка.

Последняя строка в файле содержит один символ — 0.

Формат выходных данных

Вам нужно для каждой пары спичек сгенерировать строчку `CONNECTED`, если данные спички связаны, и `NOT CONNECTED`, наоборот. Будем считать, что спичка связана сама с собой.

Пустые строки между ответами не вставлять.

Пример

input.txt	output.txt
7	CONNECTED
1 6 3 3	NOT CONNECTED
4 6 4 9	CONNECTED
4 5 6 7	CONNECTED
1 4 3 5	NOT CONNECTED
3 5 5 5	CONNECTED
5 2 6 3	
5 4 7 2	
1 4	
1 6	
3 3	
6 7	
2 3	
1 3	
0 0	
0	

Задача 7. Кратчайшие пути

Источник: повышенной сложности
Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Дан взвешенный ориентированный граф и вершина s в нём. 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, рёбер и номер выделенной вершины s соответственно ($2 \leq n \leq 2000$, $1 \leq m \leq 5000$).

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

input.txt	output.txt
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	

Замечания

Обратите внимание, что числа порядка 10^{15} не влезают в тип `int`.

Наличие *кратных* рёбер в графе означает, что между двумя вершинами может быть несколько рёбер.

Петлями называются рёбра, которую ведут из вершины в неё саму.

Задача 8. Транзитивное замыкание+

Источник: повышенной сложности
Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 1 секунда
Ограничение по памяти: разумное

Дано бинарное отношение R над множеством чисел $X = \{1, 2, 3, \dots, N\}$. Требуется найти его рефлексивно-транзитивное замыкание.

Формат входных данных

В первой строке записано одно целое число N — размер множества ($1 \leq N \leq 2000$). Далее идёт N строк по N символов в каждой, задающие отношение R . j -ый символ i -ой строки равен 1, если пара $(i, j) \in R$ (т.е. лежит в отношении R), и равен 0 в противном случае.

Формат выходных данных

Выведите N строк по N символов в каждой — рефлексивное и транзитивное замыкание отношения R , описанное в том же формате, что и исходное отношение R во входных данных.

Пример

input.txt	output.txt
5	10011
00001	11011
10010	11111
01101	10011
10000	10011
00011	