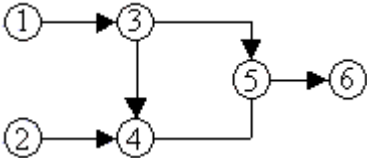


Шагалиев Владислав	Винни-Пух ходит в гости. У Винни-Пуха была прекрасная традиция – по праздникам обходить всех своих друзей. Чтобы никого не забыть он составил себе карту, где крестиками пометил дома друзей, а линиями – тропинки между домами. Так как лес был волшебным, то наличие тропинки от i-го дома к j-му не гарантирует, что по ней можно пройти от j-го дома к i-му. Однажды Винни-Пух задумался над тем, может ли он, выйдя из своего дома и побывав у каждого своего друга по одному разу, вернуться домой? Помогите, если это возможно, построить кратчайший маршрут обхода друзей (при условии, что у каждого в гостях он побывает только один раз), если известна длина каждой тропинки в шагах Винни-Пуха. <i>Технические требования:</i> входной файл INP1.TXT, выходной файл OUT1.TXT. <i>Формат входных данных:</i> входной файл содержит последовательность строк: N – количество друзей Винни Пуха ($N \leq 20$). В последующих строках записаны через пробел три целых числа: первые два – это упорядоченная пара, соответствующая номерам домов, которые соединены тропинкой; третье число – длина тропинки в шагах Винни-Пуха. Считается, что домик Винни Пуха имеет номер, равный $(N+1)$. <i>Формат выходных данных:</i> 1-я строка: Да / Нет – <i>ответ, существует ли такой маршрут.</i> 2-я строка: 0, <i>если такой маршрут не существует или</i> $R_{N+1}, \dots, R_i, \dots, R_j$ – <i>последовательность номеров домов, через которые проходит маршрут.</i> 3-я строка: 0, <i>если такой маршрут не существует или</i> K – <i>длина маршрута (в шагах Винни Пуха).</i> <i>Пример входного файла:</i> <pre>3 1 2 1 1 2 2 2 1 3 2 3 3 3 4 2 4 1 4</pre> <i>Пример выходного файла:</i> <pre>Да 4 1 2 3 4 10</pre>
Олонов Юрий	“Классификация кластеров”. Военный укрепленный район представляет собой простую квадратную решетку размером $N * N$ с расстоянием между узлами, равным 1. В некоторых узлах этой решетки находятся доты. Будем называть кластером ранга K группу из K дотов, такую, что расстояния между любыми дотами внутри этой группы меньше, чем расстояния между любым дотом, входящим в данную группу, и каждым дотом вне ее. Будем называть кластер ортогональным, если он не входит в состав кластера более высокого ранга. Максимальный ранг кластера ограничим числом 3. Написать программу, которая для заданных N, числа дотов M и их координат (X, Y) проводит классификацию дотов по ортогональным кластерам, выводит число ортогональных кластеров 3-го и 2-го рангов. В программе предусмотреть корректность ввода исходных данных. <u>Пример работы правильно работающей программы</u> Размер решетки $N=?$ 100 Число дотов $M=?$ 6 Координаты 1-го дота? 0 0 Координаты 2-го дота?

	2 1 Координаты 3-го дота? 4 0 Координаты 4-го дота? 2 99 Координаты 5-го дота? 5 90 Координаты 6-го дота? 50 50 Число ортогональных кластеров 3-го ранга равно 1 Число ортогональных кластеров 2-го ранга равно 1
Михайлова Дарья	Винни-Пух ищет ключи. У Винни-Пуха была прекрасная традиция – по праздникам обходить всех своих друзей. Чтобы никого не забыть он составил себе карту, где крестиками пометил дома друзей, а линиями – тропинки между домами. Так как лес был волшебным, то наличие тропинки от i-го дома к j-му не гарантирует, что по ней можно пройти от j-го дома к i-му. Однажды Винни-Пух потерял ключи от своего домика, пока «ходил по гостям» и задумался над тем, может ли он, выйдя из своего дома, пройти по каждой тропинке ровно один раз и снова вернуться домой? Помогите, если это возможно, построить такой маршрут обхода всех тропинок. <i>Технические требования:</i> входной файл INP1.TXT, выходной файл OUT1.TXT. <i>Формат входных данных:</i> входной файл содержит последовательность строк: N – количество друзей Винни Пуха ($N \leq 20$). В последующих строках записаны через пробел два целых числа, являющиеся упорядоченной парой, соответствующей номерам домов, которые соединены тропинкой. Считается, что домик Винни Пуха имеет номер, равный $(N+1)$. <i>Формат выходных данных:</i> 1-я строка: Да / Нет – <i>ответ, существует ли такой маршрут.</i> 2-я строка: 0, если такой маршрут не существует или $R_{N+1}, \dots, R_i, \dots, R_j$ – <i>последовательность номеров домов, через которые проходит маршрут.</i> <i>Пример входного файла:</i> 3 1 2 1 2 2 1 2 3 3 4 4 1 <i>Пример выходного файла:</i> Да 4 1 2 1 2 3 4
Корнеев Игорь	Ладьи. Дана шахматная доска размером $N \times N$ ($1 \leq N \leq 1000$), в некоторых клетках которой стоят ладьи. Будем считать, что две ладьи одинакового цвета бьют друг друга, если: 1. они стоят на одной горизонтали или вертикали; 2. между ними нет других ладей. Требуется написать программу, которая выполняет раскраску расположенных на шахматной доске ладей наименьшим количеством цветов таким образом, чтобы одноцветные ладьи не били друг друга. <i>Входной файл</i> содержит следующую последовательность строк. В первой строке записано целое число N. Каждая из последующих N строк служит для описания расположения ладей на шахматной доске по горизонталям. Клетка с ладьей обозначается буквой 'Л', без ладьи – буквой 'П'. Символы в одной строке разделены пробелами.

	Выходной файл должен содержать одно целое число – искомое количество цветов. Пример файла входных данных: <pre>3 ЛЛЛЛ ПЛП ЛПП Пример файла выходных данных: 2</pre>
Бакулева Екатерина	"Тату, тату, наши сети притащили..." Сетью называется совокупность точек (узлов), некоторые из которых соединены между собой стрелками. Сети, состоящие из n узлов, можно сопоставить две квадратные матрицы порядка n: матрицу соединений и матрицу связей. Элемент матрицы соединений A_{ij} равен 1, если сеть содержит стрелку, ведущую из узла i в узел j, и 0 в противном случае ($i, j = 1, \dots, n$). Элемент B_{ij} матрицы связей равен 1, если из узла i можно попасть в узел j, двигаясь по стрелкам, и 0 в противном случае. Так, для сети, изображенной на рисунке, матрицы соединений и связей имеют вид:  <pre> 001000 001111 000100 000100 000110 000111 000101 000101 000000 000000 </pre> Дана матрица соединений некоторой сети из n узлов; получить матрицу связей этой сети.
Солодникова Юлия	“Лабиринт”. Путник попал в лабиринт, состоящий из одинаковых комнат, каждая из которых может иметь от одной до четырех дверей в соседние комнаты. Он долго блуждал по нему и наконец нашел выход. Путник составил описание своего маршрута, обозначая в каждой комнате направления движения соответственно буквами N (север), E (восток), S (юг), W (запад). <u>Задание</u> Написать программу, определяющую по известному маршруту путника кратчайший путь от исходной точки до выхода. <u>Технические требования</u> Маршрут путника записывается в виде последовательности менее 80 символов N, E, S, W.
Ларионов Сергей	Шахматная доска с удаленными клетками. Рассмотрим шахматную доску порядка N, у которой удалено несколько клеток (не более N^2-2). Клетки называются соседними, если имеют общую сторону. Назовем маршрутом ломаную, соединяющую две клетки, причем каждый отрезок ломаной соединяет центры соседних клеток. Если для любых двух неудаленных клеток существует хотя бы один маршрут, то доска называется целой. Определить, является ли заданная доска целой. <u>Технические замечания</u> $1 < N < 21$. - Если соответствующая клетка удалена, то вводится 0, иначе 1. <u>Пример работы программы</u> Введите N 4 Введите 1-ую строку 1010

	Введите 2-ую строку 1110 Введите 3-ую строку 1001 Введите 4-ую строку 1111 Целая.																						
Рассамахин Илья	Web-адреса. Дано N Web-адресов страниц Internet ($N \leq 20$). На каждой странице расположено несколько ссылок на другие адреса (в частности, таких ссылок может не быть вообще). Требуется определить минимальное число переходов, чтобы попасть с одной страницы на другую. <u>Замечания:</u> - Ссылки однонаправлены, то есть если с A есть ссылка на B, то это не значит, что с B есть ссылка на A. - Исходные данные корректны и их проверка не требуется. <u>Технические требования:</u> - Все адреса считаются занумерованными от 1 до N. - Для каждого адреса ссылки вводятся как номера соответствующих адресов. - Программа должна позволять выполнить несколько тестов при одном и том же наборе Web-адресов (без их повторного ввода). <u>Пример работы правильной программы</u> <table> <tr> <th>Программа</th><th>Пользователь</th></tr> <tr> <td>Введите количество Web-адресов</td><td>3</td></tr> <tr> <td>Укажите ссылки для первого адреса</td><td>2 3</td></tr> <tr> <td>Укажите ссылки для второго адреса</td><td>1</td></tr> <tr> <td>Укажите ссылки для третьего адреса</td><td>2</td></tr> <tr> <td>Введите номер начального адреса</td><td>1</td></tr> <tr> <td>Введите номер конечного адреса</td><td>3</td></tr> <tr> <td>Число достаточных переходов: 1. Продолжить?</td><td>Y</td></tr> <tr> <td>Введите номер начального адреса</td><td>3</td></tr> <tr> <td>Введите номер конечного адреса</td><td>1</td></tr> <tr> <td>Число достаточных переходов: 2. Продолжить?</td><td>N</td></tr> </table>	Программа	Пользователь	Введите количество Web-адресов	3	Укажите ссылки для первого адреса	2 3	Укажите ссылки для второго адреса	1	Укажите ссылки для третьего адреса	2	Введите номер начального адреса	1	Введите номер конечного адреса	3	Число достаточных переходов: 1. Продолжить?	Y	Введите номер начального адреса	3	Введите номер конечного адреса	1	Число достаточных переходов: 2. Продолжить?	N
Программа	Пользователь																						
Введите количество Web-адресов	3																						
Укажите ссылки для первого адреса	2 3																						
Укажите ссылки для второго адреса	1																						
Укажите ссылки для третьего адреса	2																						
Введите номер начального адреса	1																						
Введите номер конечного адреса	3																						
Число достаточных переходов: 1. Продолжить?	Y																						
Введите номер начального адреса	3																						
Введите номер конечного адреса	1																						
Число достаточных переходов: 2. Продолжить?	N																						
Новоструев Александр	Кластеры шашек. В произвольных клетках шахматной доски расставлены N шашек, причем не больше одной в клетке. Будем называть кластерами такие группы компактно расположенных шашек, которые группируются по правилам: - каждая шашка может входить не более чем в один кластер; - если две шашки расположены в соседних клетках (имеющих общую вершину или сторону), то они входят в состав одного кластера. Написать программу, которая для заданных числа шашек N и их координат выведет число кластеров и число одиночных шашек. Предусмотреть проверку корректности ввода исходных данных. <u>Пример правильно работающей программы</u> <table> <tr> <th>Программа</th><th>Пользователь</th></tr> <tr> <td></td><td></td></tr> <tr> <td>N=?</td><td>7</td></tr> <tr> <td></td><td></td></tr> <tr> <td>Координаты?</td><td>a1 g7 h6 h5 g4 c1 d2</td></tr> </table>	Программа	Пользователь			N=?	7			Координаты?	a1 g7 h6 h5 g4 c1 d2												
Программа	Пользователь																						
N=?	7																						
Координаты?	a1 g7 h6 h5 g4 c1 d2																						

	Число кластеров равно 2	
	Число отдельных шашек равно 1	
Жуков Александр	«Автобусные маршруты». Для решения транспортной проблемы в некотором городе до недавнего времени использовались N ($N \leq 100$) автобусных маршрутов. Каждый маршрут начинался на одной из M ($M \leq 200$) площадей и там же заканчивался. В процессе проезда по маршруту автобус мог несколько раз проезжать одну и ту же площадь, но не мог проезжать более одного раза по одной и той же улице в том же самом направлении. В определенный момент местные власти решили сократить количество автобусных маршрутов в городе до одного. По их мнению, должен был остаться лишь один маршрут, который проходил бы по всем улицам, по которым раньше проходили автобусные маршруты, причем в том же направлении. Если по каким-либо улицам автобусы ездили в обоих направлениях, то и новый маршрут должен проходить по этим улицам в обоих направлениях. По тем улицам и в тех направлениях, по которым раньше автобусы не ездили, новый маршрут проходить не должен. Кроме того, новый маршрут должен удовлетворять всем требованиям старых маршрутов, то есть начинаться и заканчиваться на одной площади, не проходить более одного раза по одной и той же улице в том же самом направлении. Требуется написать программу, которая для заданных исходных данных определяет требуемый местными властями автобусный маршрут. <u>Формат входных данных:</u> Первая строка содержит число N — количество автобусных маршрутов. Далее следует N строк, каждая из которых служит для описания соответствующего автобусного маршрута и содержит сначала число k ($k \leq 1000$), определяющее количество элементов маршрута, а затем $k + 1$ чисел, задающих номера площадей, которые последовательно проезжает автобус на этом маршруте. При описании маршрута всегда задаются номера первой и последней площадей маршрута, причем они всегда совпадают. <u>Формат выходных данных:</u> Либо описание нового маршрута в указанном выше формате, либо одно число 0, если организовать требуемый маршрут невозможно. <u>Пример файла входных данных:</u> <pre>3 6 1 2 5 7 5 2 1 4 1 4 7 4 1 5 2 3 6 5 4 2</pre> <u>Пример файла выходных данных</u> <pre>15 2 5 4 2 3 6 5 7 4 1 2 1 4 7 5 2</pre>	
Салтыков Александр	Мальчик-с-пальчик и черви-убийцы. Жил-был мальчик-с-пальчик. Жил он правда в детском приюте. Жизнь в детском приюте совсем не сказка, равно как и эта задача. Решил мальчик-с-пальчик прогуляться одним прекрасным вечером за апельсинами. Однако вечером коварные служительницы детприютовского порядка запирают все двери и окна на засов и выбраться из дома никто не может. Но мальчик-с-пальчик — ребенок особенный. Он не только мал, но и умен. Он знает, что подвал никогда не закрывается. Решил он пробраться в подвал — мало ли, быть может там будет какая-то лазейка. Подходит он к двери, ведущей в подвал, и видит кровавую надпись «Кто сюда войдет, тому гибели не миновать». «Ничего, и не в таких переделках бывали» - подумал мальчик-с-пальчик и открыл дверь. Навстречу ему пахнуло каким-то странным	

	запахом. Ему вспомнилась ферма дяди Пети, на которой он иногда бывал до попадания в приют. На этой ферме веяло таким же запахом из мясницкой. «Апельсины важнее» – подумал мальчик-с-пальчик и вошел. Поискал он по подвалу лазеек, побродил, походил и вдруг наткнулся на какой-то тоннель в землистом полу. И подумал мальчик-с-пальчик, что быть может, вот здесь то он и сможет выбраться наружу и сбежать за своими любимыми апельсинами. Он еще и не подозревал, какие опасности его там ждут. Влез он в тоннель и побежал вдоль него. А тоннель то за ним вдруг ни с того, ни с сего завалило. Оказывается, тоннель был очень маленьким, и никто, кроме мальчика-с-пальчика его разглядеть не мог. Ну вот, кто-то неосторожно взял, да и наступил на эту дырочку в полу – ее землей и засыпало. Теперь у него остался лишь один выход – по ту сторону туннеля. К счастью, у мальчика-с-пальчика оказался в кармане коммуникатор с включенной функцией GPS (системой спутниковой навигации). А на экране коммуникатора вдруг оказалось много движущихся точек! И точки эти двигались с поразительной скоростью! Оказалось, что тоннель то был не простым. Он был вырыт злыми червями-убийцами. Когда вход в тоннель обрушился, звук разошелся по всем туннелям и разбудил спящих в своих логовах червей-убийц. Под землей существует несколько логов червей. В каждом логове бесчисленное количество червяков. Пробудившись, они расползаются во все туннели. Скорость червей, к счастью, в 2 раза меньше скорости мальчика-с-пальчика. Вам нужно определить маршрут, по которому нужно двигаться мальчику-с-пальчику, чтобы не оказаться съеденным. Тоннели задаются в виде графа (все читается из файла). Также в файле задается местоположение мальчика-с-пальчика и точка выхода на поверхность, а также местоположения логов червей. Однако бывает так, что иногда в некоторых соединительных точках туннелей возьми да появиись новое логово червей. И оттуда они тоже начинают расползаться во все стороны. (Быть может, у них тоже есть передовая техника – червепорты?) Также мальчик-с-пальчик иногда может произвести точечный удар по червям. Поскольку тоннели находятся на небольшой глубине, то мальчик-с-пальчик через свой коммуникатор может вызвать элитный эскадрон птиц летучих, которые обладают способностью проклевывать землю. Мальчик-с-пальчик указывает соединительную точку туннелей, и в этот же момент эскадрон сразу же проклевывает червей во всех туннелях, инцидентных точке удара. Изначально данные хранятся в файле, затем вводятся команды от пользователя системы контроля.
Гильманов Артур	Бедный машинист. На диком-диком Западе местные ковбои любят частенько грабить поезда. И вот, в один прекрасный день (правда, прекрасным для машиниста его назвать, наверное, язык не повернется) решили разбойники захватить поезд. Поезд отъехал из некоторого городка и тут же был захвачен бандитами. На диком-диком Западе все не как у людей, царит полный беспредел, аж до такой степени, что существует бандитский город. Туда бандиты пригоняют поезда с золотом, в нем пассажиры поездов нещадно эксплуатируются в качестве рабов. Бандиты решили направить поезд в город бандитов, и сказали бедному машинисту: «Чем быстрее доставишь нас до города бандитов, тем больше шансов у тебя сохранить свою рабскую жизнь». Бандиты – народ самоуверенный, и они думают, что машинист поедет кратчайшим путем. Но машинист – еще тот кадр. Сложно на диком-диком Западе найти законопослушного человека, но машинист оказался именно таким. Поэтому на каждой станции ему приходится делать выбор между совестью и пулей в голове. Помимо города бандитов есть и его антипод – город полисменов. В городе полисменов царит полный порядок, и именно туда стремится попасть бедный машинист. На каждой второй станции бандиты теряют бдительность, отводят пушку от головы машиниста. И тут-то совесть берет верх – машинист поворачивает на самый короткий путь к городу полисменов. Но бандиты четко знают маршрут к городу бандитов и поэтому замечают малейшее отклонение от курса. Поэтому если машинист поворачивает на станции на дорогу, которая не находится на кратчайшем пути от этой станции до города бандитов, они снова приставляют пушку к голове машиниста, и тогда машинист продолжает подчиняться. Но с каждым таким эпизодом количество шансов машиниста выжить уменьшается.

Изначально бандиты дали ему возможность совершить определенное максимальное количество ошибок. Если машинист превысит этот предел, отведенный ему срок жизни совпадет с моментом въезда в город бандитов. Нужно определить, выживет ли машинист, если будет ехать по вышеуказанным правилам, и за какое время поезд доберется до города бандитов, если вообще сможет добраться (для каждой дороги, соединяющей две станции, известно время проезда по ней: машинист – человек опытный, может вести поезд с постоянной скоростью). Но полисмены не спят, и совершают периодически облавы. К сожалению, они не знают о местоположении города бандитов и выбирают место для облавы «как попало». А именно, они выбирают некоторую дорогу и ждут там появления поезда с некоторого момента времени t_1 до момента времени t_2 . Если поезд окажется на этой дороге в указанный промежуток времени, то будет немедленно захвачен. По случайному стечению обстоятельств в этот же момент времени в голове машиниста оказывается «непонятный металлический предмет». Будем считать, что Вам известны все планы полисменов, но Вы ведь никому не расскажете о них? Ведь Вы не хотите преждевременной смерти машиниста?

Агапова
Ксения

Цветной дождь

В Банановой республике очень много холмов, соединенных мостами. На химическом заводе произошла авария, в результате чего испарилось экспериментальное удобрение "зован". На следующий день выпал цветной дождь, причем он прошел только над холмами. В некоторых местах падали красные капли, в некоторых - синие, а в остальных - зеленые, в результате чего холмы стали соответствующего цвета. Президенту Банановой республики это понравилось, но ему захотелось покрасить мосты между вершинами холмов так, чтобы мосты были покрашены в цвет холмов, которые они соединяют. К сожалению, если холмы разного цвета, то покрасить мост таким образом не удастся. Посчитайте количество таких "плохих" мостов.

Формат входных данных:

В файле INPUT.TXT в первой строке записано N ($0 < N \leq 100$) - число холмов. Далее идет матрица смежности, описывающая наличие мостов между холмами (1-мост есть, 0-нет). В последней строке записано N чисел, обозначающих цвет холмов: 1 - красный; 2 - синий; 3 - зеленый.

Формат выходных данных:

В файл OUTPUT.TXT вывести количество "плохих" мостов.

Пример

input.txt	output.txt
<pre> 7 0 1 0 0 0 1 1 1 0 1 0 0 0 0 0 1 0 0 1 1 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 1 0 1 0 1 0 0 1 0 0 0 0 0 0 1 1 1 1 1 3 3 </pre>	<pre> 4 </pre>

Издевательство

Штирлиц ехал на машине, увидел голосующего Бормана, и проехал мимо. Через некоторое время он снова увидел голосующего Бормана, и снова проехал мимо. Вскоре он опять увидел голосующего Бормана.
- Издевается! - подумал Борман.
- Кольцевая! - догадался Штирлиц.

В городе N площадей. Любые две площади соединены между собой ровно одной дорогой с двусторонним движением. В этом городе живет Штирлиц. У Штирлица есть хобби - он любит воскресным утром выйти из дома, сесть в машину, выбрать какой-нибудь кольцевой маршрут, проходящий ровно по трем площадям (то есть сначала он едет с какой-то площади на какую-то другую, потом - на третью, затем возвращается на начальную, и опять едет по этому маршруту). Он воображает, что где-то на этом пути стоит Борман. И так вот ездит Штирлиц все воскресенье, пока голова не закружится, и радуется...

Естественно, что Штирлицу хочется проезжать мимо точки, в которой, как он воображает, стоит Борман, как можно чаще. Для этого, естественно, выбранный Штирлицем маршрут должен быть как можно короче. Напишите программу, которая выберет оптимальный для Штирлица маршрут.

Формат входных данных:

Во входном файле INPUT.TXT записано сначала число N ($3 \leq N \leq 100$), а затем матрица NxN расстояний между площадями (число в позиции i, j обозначает длину дороги, соединяющей i-ю и j-ю площади). Все числа в матрице (кроме стоящих на главной диагонали) - натуральные, не превышающие 1000. Матрица симметрична относительно главной диагонали, на главной диагонали стоят 0.

Формат выходных данных:

В выходной файл OUTPUT.TXT выведите номера площадей в оптимальном маршруте. Если маршрутов несколько, выведите любой из них.

Пример

input.txt	output.txt
5 0 20 10 30 40 20 0 30 1 2 10 30 0 40 1000 30 1 40 0 21 40 2 1000 21 0	4 5 2