

Задача А. Обход в ширину

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Дан неориентированный граф. В нём необходимо найти расстояние от одной заданной вершины до другой.

Формат входных данных

В первой строке входного файла содержатся три натуральных числа N , S и F ($1 \leq S, F \leq N \leq 100$) — количество вершин в графе и номера начальной и конечной вершин соответственно. Далее в N строках задана матрица смежности графа. Если значение в j -м элементе i -й строки равно 1, то в графе есть направленное ребро из вершины i в вершину j .

Формат выходных данных

В единственной строке должно находиться минимальное расстояние от начальной вершины до конечной. Если пути не существует, выведите 0.

Пример

стандартный ввод	стандартный вывод
5 4 2 0 1 0 1 1 1 0 1 0 0 0 1 0 1 0 1 0 1 0 0 1 0 0 0 0	2

Задача В. Приключения шахматного коня

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

На шахматной доске $N \times N$ в клетке (x_1, y_1) стоит голодный шахматный конь. Он хочет попасть в клетку (x_2, y_2) , где растет вкусная шахматная трава. Какое наименьшее количество ходов он должен для этого сделать?

Формат входных данных

На вход программы поступает пять чисел: N, x_1, y_1, x_2, y_2 ($5 \leq N \leq 20, 1 \leq x_1, y_1, x_2, y_2 \leq N$). Левая верхняя клетка доски имеет координаты $(1, 1)$, правая нижняя — (N, N) .

Формат выходных данных

В первой строке выведите единственное число K — количество посещенных клеток. В каждой из следующих K строк должно быть записано 2 числа — координаты очередной клетки в пути коня.

Пример

стандартный ввод	стандартный вывод
5	2
1 1	1 1
3 2	3 2

Задача C. Числа

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Витя хочет придумать новую игру с числами. В этой игре от игроков требуется преобразовывать четырехзначные числа не содержащие нулей при помощи следующего разрешенного набора действий:

1. Можно увеличить первую цифру числа на 1, если она не равна 9.
2. Можно уменьшить последнюю цифру на 1, если она не равна 1.
3. Можно циклически сдвинуть все цифры на одну вправо.
4. Можно циклически сдвинуть все цифры на одну влево.

Например, применяя эти правила к числу 1234 можно получить числа 2234, 1233, 4123 и 2341 соответственно. Точные правила игры Витя пока не придумал, но пока его интересует вопрос, как получить из одного числа другое за минимальное количество операций.

Формат входных данных

Во входном файле содержится два различных четырехзначных числа, каждое из которых не содержит нулей.

Формат выходных данных

Программа должна вывести последовательность четырехзначных чисел, не содержащих нулей. Последовательность должна начинаться первым из данных чисел и заканчиваться вторым из данных чисел, каждое последующее число в последовательности должно быть получено из предыдущего числа применением одного из правил. Количество чисел в последовательности должно быть минимально возможным.

Пример

стандартный ввод	стандартный вывод
1234	1234
4321	2234
	3234
	4323
	4322
	4321

Задача D. Эвакуация

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Одна из Сверхсекретных организаций, чье название мы не имеем право разглашать, представляет собой сеть из N подземных бункеров, соединенных равными по длине туннелями, по которым из любого бункера можно добраться до любого другого (не обязательно напрямую). Связь с внешним миром осуществляется через специальные засекреченные выходы, которые расположены в некоторых из бункеров. Организации понадобилось составить план эвакуации персонала на случай экстренной ситуации. Для этого для каждого из бункеров необходимо узнать, сколько времени потребуется для того, чтобы добраться до ближайшего из выходов. Вам, как специалисту по таким задачам, поручено рассчитать необходимое время для каждого из бункеров по заданному описанию помещения Сверхсекретной организации. Для вашего же удобства бункеры занумерованы числами от 1 до N .

Формат входных данных

В первой строке записано число N , во второй — число K ($1 \leq N \leq 100\,000$, $1 \leq K \leq N$) — количество бункеров и количество выходов соответственно. Далее через пробел записаны K различных чисел от 1 до N , обозначающих номера бункеров, в которых расположены выходы. Потом идёт целое число M ($1 \leq M \leq 100\,000$) — количество туннелей. Далее вводятся M пар чисел — номера бункеров, соединенных туннелем. По каждому из туннелей можно двигаться в обе стороны. В организации не существует туннелей, ведущих из бункера в самого себя, зато может существовать более одного туннеля между парой бункеров.

Формат выходных данных

В первой строке выведите N чисел, разделённых пробелом — для каждого из бункеров минимальное время, необходимое чтобы добраться до выхода. Считайте, что время перемещения по одному туннелю равно 1. Во второй строке выведите N чисел — для каждого бункера номер ближайшего бункера с выходом, если таких несколько выведите бункер с наименьшим номером.

Пример

стандартный ввод	стандартный вывод
3	1 0 1
1	2 2 2
2	
3	
1 2	
3 1	
2 3	

Задача Е. Двудольный граф

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Двудольным называется неориентированный граф $\langle V, E \rangle$, вершины которого можно разбить на два множества L и R , так что $L \cap R = \emptyset$, $L \cup R = V$ и для любого ребра $(u, v) \in E$ либо $u \in L, v \in R$, либо $v \in L, u \in R$.

Дан неориентированный граф. Требуется проверить, является ли он двудольным.

Формат входных данных

Первая строка входного файла содержит два натуральных числа n и m — количество вершин и ребер графа соответственно ($1 \leq n \leq 100\,000$, $0 \leq m \leq 200\,000$).

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

Формат выходных данных

В единственной строке выходного файла выведите «YES», если граф является двудольным и «NO» в противном случае.

Если граф двудольный, выведите правильное разбиение на два множества: для каждой вершины выведите 1, если она входит в множество L , и 2 иначе.

Примеры

стандартный ввод	стандартный вывод
4 4 1 2 1 3 2 4 4 2	YES 1 2 2 1
3 3 1 2 2 3 3 1	NO

Задача F. Кратчайший путь

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Дан ориентированный взвешенный граф без ребер отрицательного веса. Найдите кратчайшее расстояние от одной заданной вершины до другой.

Формат входных данных

В первой строке входного файла три числа: n , s и f ($1 \leq n \leq 2000, 1 \leq s, f \leq n$), где n — количество вершин графа, s — начальная вершина, а f — конечная. 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. Вес каждого ребра не превышает 10^9 . На главной диагонали матрицы всегда нули.

Формат выходных данных

Вывести искомое расстояние или -1 , если пути между указанными вершинами не существует.

Пример

стандартный ввод	стандартный вывод
3 1 2 0 -1 2 3 0 -1 -1 4 0	6

Задача G. Кратчайший путь – 2

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Дан неориентированный связный взвешенный граф. Найдите кратчайшее расстояние от первой вершины до всех вершин.

Формат входных данных

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 .

Формат выходных данных

Выведите n чисел — для каждой вершины кратчайшее расстояние до нее.

Пример

стандартный ввод	стандартный вывод
4 5 1 2 1 1 3 5 2 4 8 3 4 1 2 3 3	0 1 4 5

Задача Н. Флойд

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Полный ориентированный взвешенный граф задан матрицей смежности. Постройте матрицу кратчайших путей между его вершинами. Гарантируется, что в графе нет циклов отрицательного веса.

Формат входных данных

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.

Формат выходных данных

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 .

Пример

стандартный ввод	стандартный вывод
4	0 5 7 13
0 5 9 100	12 0 2 8
100 0 2 8	11 16 0 7
100 100 0 7	4 9 11 0
4 100 100 0	

Задача I. Лабиринт знаний

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

В Летней Компьютерной Школе (ЛКШ) построили аттракцион "Лабиринт знаний". Лабиринт представляет собой N комнат, занумерованных от 1 до N , между некоторыми из которых есть двери. Когда человек проходит через дверь, показатель его знаний изменяется на определенную величину, фиксированную для данной двери. Вход в лабиринт находится в комнате 1, выход – в комнате N . Каждый ученик проходит лабиринт ровно один раз и попадает в ту или иную учебную группу в зависимости от количества набранных знаний (при входе в лабиринт этот показатель равен нулю). Ваша задача показать наилучший результат.

Формат входных данных

Первая строка входных данных содержит целые числа N ($1 \leq N \leq 2000$) – количество комнат и M ($1 \leq M \leq 10000$) – количество дверей. В каждой из следующих M строк содержится описание двери – номера комнат, из которой она ведет и в которую она ведет (через дверь можно ходить только в одном направлении), а также целое число, которое прибавляется к количеству знаний при прохождении через дверь (это число по модулю не превышает 10000). Двери могут вести из комнаты в нее саму, между двумя комнатами может быть более одной двери.

Формат выходных данных

Выведите " :)" – если можно получить неограниченно большой запас знаний, " (" – если лабиринт пройти нельзя, и максимальное количество набранных знаний в противном случае.

Пример

стандартный ввод	стандартный вывод
2 2 1 2 3 1 2 7	7

Задача J. Транспортировка

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Компания “Яндекс” решила подарить всем школьникам и преподавателям Школы Олимпиадной Подготовки оригинальные кружки. К сожалению, количество необходимых кружек оказалось столь велико, что изготовитель доставил кружки в офис Яндекса в самый последний момент. До открытия смены в Иннополисе осталось всего 24 часа.

О плачевном состоянии дорог по пути до города ходят легенды — в частности, на многих разбитых дорогах действует ограничение на вес автомобиля. Соответственно, от нагруженности машины зависит возможность воспользоваться тем или иным маршрутом, тяжёлой машине может потребоваться ехать в обход.

Уже совершенно очевидно, что все кружки не поспеют к открытию. Чтобы спасти ситуацию, отвезите первым рейсом максимально возможное количество кружек успев до начала открытия смены.

Формат входных данных

В первой строке находятся целые числа n ($2 \leq n \leq 500$) и m — количество городов и количество двусторонних дорог, соответственно.

В следующих m строках описываются дороги.

В каждой строке находятся целые числа a_i, b_i, t_i, w_i — соответственно два города, ею соединяемые, время на проезд по ней в минутах и ограничение на вес автомобиля в граммах ($t_i \leq 1440$, $w_i \leq 10^9$, $1 \leq a_i, b_i \leq n$)

Между каждой парой городов есть не более одной дороги.

Кроме того, известно, что офис Яндекса имеет номер 1, а Иннополис — номер n , одна кружка весит 100 грамм, а пустой грузовик — 3 тонны.

Формат выходных данных

Выведите одно число — максимальное количество кружек, которое можно привезти, потратив не более 24 часов.

Пример

стандартный ввод	стандартный вывод
3 3 1 2 10 3000220 2 3 20 3000201 1 3 1 3000099	2

Задача K. Pink Floyd

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Группа Pink Floyd собирается дать новый концертный тур по всему миру. По предыдущему опыту группа знает, что солист Роджер Уотерс постоянно нервничает при перелетах. На некоторых маршрутах он теряет вес от волнения, а на других — много ест и набирает вес.

Известно, что чем больше весит Роджер, тем лучше выступает группа, поэтому требуется спланировать перелеты так, чтобы вес Роджера на каждом концерте был максимально возможным. Группа должна посещать города в том же порядке, в котором она дает концерты. При этом между концертами группа может посещать промежуточные города.

Формат входных данных

Первая строка входного файла содержит три натуральных числа n , m и k — количество городов в мире, количество рейсов и количество концертов, которые должна дать группа соответственно ($n \leq 100, m \leq 10^4, 2 \leq k \leq 10^4$). Города пронумерованы числами от 1 до n . Следующие m строк содержат описание рейсов, по одному на строке. Рейс номер i описывается тремя числами b_i, e_i и w_i — номер начального и конечного города рейса и предполагаемое изменение веса Роджера в миллиграммах ($1 \leq b_i, e_i \leq n, -10^5 \leq w_i \leq 10^5$). Последняя строка содержит числа a_1, a_2, \dots, a_k — номера городов, в которых проводятся концерты. В начале концертного тура группа находится в городе a_1 . Гарантируется, что группа может дать все концерты.

Формат выходных данных

Первая строка выходного файла должна содержать число s — количество рейсов, которые должна сделать группа. Вторая строка должна содержать s чисел — номера используемых рейсов. Если существует такая последовательность маршрутов между концертами, что Роджер будет набирать вес неограниченно, то первая строка выходного файла должна содержать строку «infinitely kind».

Примеры

стандартный ввод	стандартный вывод
4 8 5 1 2 -2 2 3 3 3 4 -5 4 1 3 1 3 2 3 1 -2 3 2 -3 2 4 -10 1 3 1 2 4	6 5 6 5 7 2 3
4 8 5 1 2 -2 2 3 3 3 4 -5 4 1 3 1 3 2 3 1 -2 3 2 -3 2 4 10 1 3 1 2 4	infinitely kind

Задача L. Кратчайший путь длины K

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 4 секунды
Ограничение по памяти: 256 мегабайт

Дан ориентированный граф. Найдите кратчайшие пути, состоящие из K рёбер, от S до всех вершин.

Формат входных данных

В первой строке дано целых четыре целых числа: $1 \leq N, M \leq 10^4$ — количества вершин и рёбер, $0 \leq K \leq 100$ — количество рёбер 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 ($1 \leq a_i, b_i \leq N, -10^5 \leq w \leq 10^5$).

Формат выходных данных

Выведите ровно N чисел по одному в строке. i -е число — длина минимального пути из ровно K рёбер из S в i , или -1 , если пути не существует.

Примеры

стандартный ввод	стандартный вывод
3 3 1 1 1 2 100 2 3 300 1 3 2	-1 100 2
3 3 2 1 1 2 100 2 3 300 1 3 2	-1 -1 400

Задача М. Цивилизация

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Карта мира в компьютерной игре “Цивилизация” версии 1 представляет собой прямоугольник, разбитый на квадратики. Каждый квадратик может иметь один из нескольких возможных рельефов, для простоты ограничимся тремя видами рельефов - поле, лес и вода. Поселенец перемещается по карте, при этом на перемещение в клетку, занятую полем, необходима одна единица времени, на перемещение в лес - две единицы времени, а перемещаться в клетку с водой нельзя.

У вас есть один поселенец, вы определили место, где нужно построить город, чтобы как можно скорее завладеть всем миром. Найдите маршрут переселенца, приводящий его в место строительства города, требующий минимального времени. На каждом ходе переселенец может перемещаться в клетку, имеющую общую сторону с той клеткой, где он сейчас находится.

Формат входных данных

Во входном файле записаны два натуральных числа N и M , не превосходящих 500 - размеры карты мира (N - число строк в карте, M - число столбцов). Затем заданы координаты начального положения поселенца x и y , где x - номер строки, y - номер столбца на карте ($1 \leq x \leq N, 1 \leq y \leq M$), строки нумеруются сверху вниз, столбцы - слева направо. Затем аналогично задаются координаты клетки, куда необходимо привести поселенца.

Далее идет описание карты мира в виде N строк, каждая из которых содержит M символов. Каждый символ может быть либо “.” (точка), обозначающим поле, либо “F”, обозначающим лес, либо “W”, обозначающим воду. Гарантируется, что начальная и конечная клетки пути переселенца не являются водой.

Формат выходных данных

В первой строке выходного файла выведите количество единиц времени, необходимое для перемещения поселенца (перемещение в клетку с полем занимает 1 единицу времени, перемещение в клетку с лесом - 2 единицы времени). Во второй строке выходного файла выведите последовательность символов, задающих маршрут переселенца. Каждый символ должен быть одним из четырех следующих: “N” (движение вверх), “E” (движение вправо), “S” (движение вниз), “W” (движение влево). Если таких маршрутов несколько, выведите любой из них.

Если дойти из начальной клетки в конечную невозможно, выведите число -1.

Пример

input.txt	output.txt
4 8 1 1 4 8FFFF .WWWWW. .W..F... ...FFFF.	13 SSSEENEEEEES