

Задача А. От списка ребер к матрице смежности, ориентированный вариант

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

Простой ориентированный граф задан списком ребер, выведите его представление в виде матрицы смежности.

Формат входных данных

На вход программы поступают числа n ($1 \leq n \leq 100$) — количество вершин в графе и m ($1 \leq m \leq n(n-1)$) — количество ребер. Затем следует m пар чисел — ребра графа.

Формат выходных данных

Выведите матрицу смежности заданного графа.

Пример

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

Задача В. BFS на очереди

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

Вам дан граф неориентированный невзвешенный G на n вершин и m ребер и вершина s . Найдите расстояния от вершины s до всех остальных вершин.

Формат входных данных

В первой строке вводятся три числа $1 \leq n, m \leq 10^5, 1 \leq s \leq n$ — количество вершин, количество ребер и стартовая вершина соответственно.

В следующих m строках вводится по два числа $1 \leq a_i, b_i \leq n$ — ребра графа G .

Формат выходных данных

В единственной строке выведите n числе, где i -е равно расстоянию от вершины s до i , если путь от s до i существует, и -1 иначе.

Пример

стандартный ввод	стандартный вывод
10 9 1 1 2 1 3 1 4 2 3 8 3 7 3 3 6 2 5 6 9	0 1 1 1 2 2 2 2 3 -1

Задача С. Дейкстра с восстановлением пути

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

Вам дан взвешенный неориентированный граф G на n вершин и m ребер и вершины s и t .
Найдите кратчайший путь от вершины s до t .

Формат входных данных

В первой строке вводятся четыре числа $1 \leq n, m \leq 2 \cdot 10^5, 1 \leq s \leq n, 1 \leq t \leq n$ — количество вершин, количество ребер, номер стартовой и конечной вершины.

В следующих m строках вводятся по три числа $1 \leq a_i, b_i \leq n, 1 \leq c_i \leq 10^9$ — ребра графа.

Формат выходных данных

Если пути из s в t не существует выведите единственное число -1 .

Иначе выведите в первой строке число d , равное расстоянию от вершины s до t , и число k , равное числу вершин в одном из кратчайших путей из s в t . Во второй строке выведите k чисел — вершины, через которые проходит один из кратчайших путей из s в t .

Пример

стандартный ввод	стандартный вывод
5 7 2 5 1 2 1 1 3 4 1 5 5 2 3 1 3 4 2 3 5 3 4 5 7	4 3 2 3 5

Задача D. Два коня путешественника

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

На шахматном поле размером $n \times n$, на клетках (x_1, y_1) и (x_2, y_2) стоят два одиноких коня-путешественника. Они хотят перестать быть одинокими конями. Для этого каждый из них хочет совершить одновременно с напарником ровно по x ходов, так чтобы оказаться в одной клетки со вторым конем-путешественником. При каком наименьшем x это возможно?

Формат входных данных

В первой строке вводится одно число $1 \leq n \leq 1000$ — размер шахматного поля.
В второй строке вводятся два числа $1 \leq x_1, y_1 \leq n$ — координаты первого коня.
В третьей строке вводятся два числа $1 \leq x_2, y_2 \leq n$ — координаты второго коня.

Формат выходных данных

В единственной строке выведите минимальный x , при котором осуществимо задуманное конями. Если такого x не существует, выведите -1 .

Пример

стандартный ввод	стандартный вывод
8 1 1 1 3	1

Задача Е. Водопады Бали

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	32 мегабайта

На Бали есть несколько водопадов, пронумерованных целыми числами от 1 до N . Между некоторыми парами водопадов проложены узкие тропинки. Всего тропинок M , и j -я из них соединяет водопады a_j и b_j . Проехать по j -й тропинке в любую сторону можно ровно за L_j минут.

Вано решил покататься на байке по заранее известному маршруту: он задан последовательностью водопадов $p_1, p_2, \dots, p_G$, которые Вано проезжает по порядку, переезжая между соседними водопадами по соединяющей их тропинке. Вано выезжает из водопада p_1 в момент времени 0 и едет без остановок. На друзей он внимания не обращает: на тропинку он въезжает в свой момент, даже если по ней уже кто-то едет.

А вот его друг Алексей очень осторожен и не въезжает на тропинку, по которой в данный момент едет Вано. Пусть Вано въезжает на тропинку длины L в момент s , где s — суммарная длина всех тропинок маршрута, пройденных им до этой. Тогда Алексей не может въехать на эту тропинку (ни с какого её конца) в моменты $s, s+1, \dots, s+L-1$. Иными словами, въезжать на тропинку нельзя с момента, когда на неё въехал Вано, и пока он по ней едет; в момент, когда Вано покидает тропинку, она уже свободна. Если же Алексей въехал на тропинку раньше момента s , он спокойно доезжает до её конца, даже если Вано въезжает на ту же тропинку, пока Алексей ещё в пути. Тропинки, не входящие в маршрут Вано, всегда свободны.

Алексей стоит у водопада A в момент K и хочет как можно быстрее оказаться у водопада B . Он может ехать по тропинкам в любую сторону, но остановиться посреди тропинки не может: тропинку длины L он проезжает ровно за L минут. У водопадов Алексей может ждать любое целое число минут. Все моменты времени, в которые Алексей въезжает на тропинки, являются целыми числами.

Помогите Алексею узнать, через сколько минут после момента K он сможет оказаться у водопада B .

Формат входных данных

Первая строка содержит два целых числа N и M — количество водопадов и количество тропинок соответственно ($2 \leq N \leq 1000$, $1 \leq M \leq 10\,000$).

Вторая строка содержит четыре целых числа A , B , K и G — номер водопада, у которого стоит Алексей, номер водопада, у которого он хочет оказаться, момент времени, в который Алексей стоит у водопада A , и количество водопадов в маршруте Вано соответственно ($1 \leq A, B \leq N$, $0 \leq K \leq 1000$, $1 \leq G \leq 1000$).

Третья строка содержит G целых чисел $p_1, p_2, \dots, p_G$ — номера водопадов маршрута Вано в порядке проезда ($1 \leq p_i \leq N$). Если $G = 1$, то Вано никуда не едет и все тропинки свободны.

Далее следуют M строк, в каждой из которых записаны три целых числа a_j , b_j и L_j — номера водопадов, которые соединяет j -я тропинка, и время проезда по ней в минутах ($1 \leq a_j, b_j \leq N$, $a_j \neq b_j$, $1 \leq L_j \leq 1000$).

Гарантируется, что никакие два водопада не соединены более чем одной тропинкой, а любые два соседних водопада маршрута p_i и p_{i+1} соединены тропинкой. Гарантируется, что каждая тропинка встречается в маршруте Вано не более одного раза (в любом из двух направлений). Также гарантируется, что от водопада A можно доехать до водопада B по тропинкам.

Формат выходных данных

Выведите одно целое число — минимальное количество минут, считая от момента K , через которое Алексей может оказаться у водопада B . Если $A = B$, выведите 0.

Примеры

стандартный ввод	стандартный вывод
6 5 1 6 20 4 5 3 2 4 1 2 2 2 3 8 2 4 3 3 6 10 3 5 15	21
8 9 1 5 5 5 1 2 3 4 5 1 2 8 2 7 4 2 3 10 6 7 40 3 6 5 6 8 3 4 8 4 4 5 5 3 4 23	40

Замечание

В первом примере Вано проезжает тропинку между водопадами 3 и 2 с момента 15 до момента 23. Алексей приезжает к водопаду 2 в момент 22, ждёт одну минуту и въезжает на эту тропинку в момент 23. Ответ равен $41 - 20 = 21$.

Во втором примере Вано въезжает на тропинку между водопадами 2 и 3 в момент 8, а между водопадами 4 и 5 — в момент 41. Алексей ждёт у водопада 1 до момента 8, в момент 16 оказывается у водопада 2 и ждёт там две минуты, пока не освободится тропинка к водопаду 3. Далее он едет по водопадам 3, 6, 8 и 4 без остановок, въезжает на тропинку 4–5 в момент 40, за минуту до Вано, и прибывает в момент 45. Ответ равен $45 - 5 = 40$.

Задача F. Мастерство тракториста

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	0.5 секунд
Ограничение по памяти:	256 мегабайт

Вредный культорг Ишма Дровкин обожает срывать учебные процессы. Особенно он любит заставлять школьников проявлять физическую активность. Вот сейчас, проявив недюжинный ораторский талант, он убедил школьников угнать трактор с хутора и отправиться кататься на нём по полю. Однако, трактор, который им предстоит угнать, старше половины преподавательского состава параллели В, вместе взятой, поэтому им невозможно рулить. Он может ехать вперёд и только вперёд.

Более формально — представим поле в форме прямоугольника $H \times W$. Некоторые клетки этого поля заблокированы и по ним невозможно проехать даже на тракторе. Трактор может быть повернут вдоль направления движения по вертикали, горизонтали или диагонали.

Школьники в ЛКШ — люди не только упорные, но и развитые физически, поэтому их не смущает перспектива разворачивать трактор своими руками каждый раз, когда им требуется повернуть. Но, естественно, это не самое приятное времяпрепровождение, так что они просят вас найти маршрут, следуя по которому они будут вынуждены работать руками минимальное количество раз. Важная деталь заключается в том, что сейчас трактор валяется на поле в перевернутом состоянии, так что по меньшей мере один раз им придётся повернуть трактор в правильное положение, которое они могут выбрать удобным им образом. То, как именно будет повернут трактор в конечной точке, не имеет значения.

Формат входных данных

В первой строке входного файла заданы целые числа H, W ($1 \leq H, W \leq 1000$) — высота и ширина поля.

В последующих H строках задано поле, заблокированные клетки обозначены латинскими буквами "X", а свободные — точками.

После поля следует строка с двумя целыми числами sx, sy ($1 \leq sx \leq W, 1 \leq sy \leq H$) — координаты стартового положения трактора.

Последней строкой идут два целых числа tx, ty ($1 \leq tx \leq W, 1 \leq ty \leq H$) — координаты конечного положения трактора.

Координаты отсчитываются от нижнего левого угла поля. Стартовое положение не совпадает с конечным. Стартовая и конечная клетки не являются заблокированными.

Формат выходных данных

Если Ишма Дровкин настолько большой шутник, что пути между стартовой и конечной клетками не существует, выведите -1 . Иначе выведите единственное натуральное число — минимальное количество поворотов, которые школьники должны будут сделать по пути к конечной клетке.

Пример

стандартный ввод	стандартный вывод
5 7 XX...X X.XXX.. ..XX.X X.X...X ...XXX 1 1 6 5	3

Замечание

В примере школьники сначала перевернут трактор и развернут его направо. Потом они проедут три клетки и повернут трактор на 45 градусов против часовой стрелки, чтобы тот смотрел по диаго-

нали вверх и вправо. Затем они проедут две клетки и снова повернут трактор на 45 градусов против часовой стрелки, чтобы тот смотрел наверх. Наконец, они проедут три клетки вверх и окажутся в конечной клетке.

Задача G. Фонтан

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	32 мегабайта

Вано гулял по Ницце и вышел на променад Promenade du Paillon. Часть площади там занимает играющий фонтан, и Вано захотел как можно скорее попасть на одну особенную плитку — невысокую сухую ступеньку, на которую вода не затекает и через которую не течёт дальше.

Площадь вымощена квадратными плитками, которые образуют прямоугольник из строк и столбцов. Плитки бывают пяти видов:

- обычные плитки, по которым вода растекается свободно;
- плитки, из которых бьют струи фонтана — они залиты водой с самого начала;
- клумбы — приподнятые плитки, огороженные бортиками: вода на них не затекает, а ходить по клумбам запрещено, поэтому Вано на них не заходит;
- ступенька, на которую хочет попасть Вано;
- плитка, на которой Вано стоит в самом начале — она обычная, вода может залить и её.

Время идёт минутами. Каждую минуту одновременно происходят два события. Вано делает шаг на одну из четырёх плиток, соседних по стороне с той, на которой он стоит; стоять на месте нельзя. Вода растекается: каждая обычная плитка, соседняя по стороне с залитой плиткой, становится залитой. Плитки, до которых вода не добирается, остаются сухими.

Вано не может зайти на залитую плитку, а также на плитку, которую вода заливает в ту же минуту, когда он на неё заходит. Иными словами, зайти на плитку можно, только если она остаётся сухой и после того, как вода растеклась. Если же Вано уже стоит на плитке, которую заливает, он может уйти с неё в эту же минуту.

Помогите Вано выяснить, за какое наименьшее число минут он сможет оказаться на ступеньке.

Формат входных данных

Первая строка содержит два целых числа R и C — число строк и число столбцов площади ($1 \leq R, C \leq 50$).

Каждая из следующих R строк содержит ровно C символов, описывающих плитки этой строки площади слева направо:

- `.` — обычная плитка;
- `*` — плитка, из которой бьёт струя фонтана;
- `X` — клумба;
- `D` — ступенька;
- `S` — плитка, на которой Вано стоит в самом начале.

Гарантируется, что символы `D` и `S` встречаются на площади ровно по одному разу. Символ `*` может встречаться несколько раз, а может не встречаться вовсе.

Формат выходных данных

Выведите одно целое число — наименьшее число минут, за которое Вано может оказаться на ступеньке. Если он не может этого сделать, выведите -1 .

Примеры

стандартный ввод	стандартный вывод
3 3 D.* S.	3
3 3 D.* S	-1
3 6 D...*. .X.X.. S.	6

Замечание

В первом и третьем примерах Вано успевает дойти до ступеньки кратчайшим из безопасных путей: в первом — вдоль левого края площади, в третьем — по нижней строке влево до края, а затем вверх. Вода приходит на каждую плитку этого пути позже Вано.

Во втором примере плитка справа от ступеньки залита уже после первой минуты, а плитка под ступенькой заливается в ту же минуту, когда Вано мог бы на неё зайти, поэтому добраться до ступеньки нельзя.

Задача Н. Числа Вано

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Вано увлекается необычными числами: он считает красивыми только те, в десятичной записи которых встречаются лишь цифры 0 и 1, причём запись не начинается с нуля. Например, числа 1, 10 и 1101 красивые, а числа 2, 12 и 120 — нет.

Для натурального числа n Ваню хочет найти наименьшее положительное целое красивое число, которое делится на n , то есть равно n , умноженному на некоторое целое число.

Помогите Ваню найти такое число.

Формат входных данных

В единственной строке записано целое число n ($1 \leq n \leq 10^6$).

Формат выходных данных

Выведите одну строку — десятичную запись искомого числа целиком, без пробелов и ведущих нулей. Если такого числа не существует, выведите -1 .

Примеры

стандартный ввод	стандартный вывод
17	11101
173	1011001101
125	1000

Замечание

В первом примере число 11101 делится на 17, а меньшего красивого числа с этим свойством нет.

В третьём примере число 1000 делится на 125.

Задача I. Воспоминания Вано

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

Вано умеет перемещаться во сне по своим воспоминаниям. Во сне есть N мест, пронумерованных целыми числами от 1 до N . У Вано есть M воспоминаний, и каждое из них связывает сразу K различных мест.

Если два места встречаются вместе хотя бы в одном воспоминании, то Вано может за один шаг перейти из одного из них в другое, в любую сторону. Порядок, в котором места перечислены внутри воспоминания, значения не имеет. Одно и то же место может входить в несколько воспоминаний, а некоторые воспоминания могут полностью совпадать.

Сон начинается в месте 1. Вано хочет оказаться в месте N , побывав при этом в как можно меньшем числе мест. Места считаются вместе с первым и последним. Если $N = 1$, то место 1 является и первым, и последним, поэтому Вано побывал ровно в одном месте.

Помогите Вано узнать, в каком наименьшем числе мест ему придётся побывать.

Формат входных данных

Первая строка входных данных содержит три целых числа N , K и M — количество мест, количество мест в одном воспоминании и количество воспоминаний ($1 \leq N \leq 100\,000$, $1 \leq K \leq N$, $1 \leq M \leq 500\,000$, $K \cdot M \leq 500\,000$).

Далее следуют M строк, в каждой из которых записаны K целых чисел от 1 до N — номера мест очередного воспоминания.

Гарантируется, что числа в одной строке попарно различны.

Формат выходных данных

Выведите одно число — наименьшее количество мест, в которых побывает Вано. Если добраться из места 1 в место N невозможно, выведите -1 .

Примеры

стандартный ввод	стандартный вывод
9 3 5 1 2 3 1 4 5 3 6 7 5 6 7 6 8 9	4
15 8 4 11 12 8 14 13 6 10 7 1 5 8 12 13 6 2 4 10 15 4 5 9 8 14 12 11 12 14 3 5 6 1 13	3

Замечание

В первом примере Вано может побывать в местах 1, 3, 6 и 9.

Во втором примере подходит путь через места 1, 8 и 15.

Задача J. Вано и кофейни

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	32 мегабайта

Вано гуляет по Москве, но терпеть не может запах кофе. Карта района — прямоугольное поле из n строк и m столбцов, состоящее из клеток. Строки пронумерованы сверху вниз, а столбцы — слева направо. Клетка может быть пустой или содержать кофейню. Кроме того, на поле есть клетка старта, где Вано находится сейчас, и клетка цели, куда ему нужно попасть.

Клетки называются *соседними*, если у них есть общая сторона. За один шаг Вано переходит из своей клетки в соседнюю. Идти через клетки с кофейнями можно, и одну и ту же клетку разрешается посещать несколько раз. *Маршрут* — это последовательность клеток от клетки старта до клетки цели, в которой любые две подряд идущие клетки соседние.

Расстоянием между клетками (r_1, c_1) и (r_2, c_2) назовём величину $|r_1 - r_2| + |c_1 - c_2|$. Иными словами, это число шагов, которое нужно сделать по соседним клеткам, чтобы добраться из одной клетки в другую. Для клетки x обозначим за $d(x)$ расстояние от неё до ближайшей кофейни. Для клетки с кофейней $d(x) = 0$.

Значением маршрута назовём наименьшее из чисел $d(x)$ по всем клеткам x маршрута, включая клетку старта и клетку цели.

Помогите Вано найти наибольшее возможное значение маршрута.

Формат входных данных

Первая строка содержит два целых числа n и m — количество строк и столбцов поля ($1 \leq n, m \leq 500$).

Далее следуют n строк по m символов в каждой, без пробелов, описывающих поле. Каждый символ равен одному из следующих:

- . — пустая клетка;
- С — клетка с кофейней;
- V — клетка старта;
- Т — клетка цели.

Гарантируется, что символы V и Т встречаются на поле ровно по одному разу, а символ С — хотя бы один раз.

Формат выходных данных

Выведите одно целое число — наибольшее возможное значение маршрута.

Примеры

стандартный ввод	стандартный вывод
4 5 C.... .V... T.	2
3 5 V.... TC	1
5 6 V..... CCCC.. T.....	2

Замечание

В первом примере ответ равен расстоянию от клетки старта до единственной кофейни, а во втором — расстоянию от клетки цели до соседней кофейни.

В третьем примере кратчайший маршрут проходит через кофейню, поэтому Ваню стоит обойти ряд кофеен справа.

Задача К. Новый тоннель

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

Градостроитель Вано решает, какой новый тоннель прокопать в метро Прекрасной страны будущего. В метро есть N станций, пронумерованных целыми числами от 1 до N , и M тоннелей. Каждый тоннель соединяет две станции и допускает движение в обе стороны; по i -му тоннелю, соединяющему станции A_i и B_i , можно проехать за время C_i . Пассажиры могут пересаживаться на станциях и проезжать подряд несколько тоннелей, при этом время поездки равно сумме времён проезда по всем использованным тоннелям.

Больше всего граждан ежедневно перемещается между станциями S и T , поэтому Вано хочет, чтобы им было удобно ездить: он собирается проложить один новый тоннель между двумя различными станциями, время проезда по которому равно L , так, чтобы после этого от станции S до станции T можно было доехать за время не более K . Новый тоннель можно проложить и между станциями, которые уже соединены тоннелем — тогда между ними окажутся два тоннеля. Возможно также, что без нового тоннеля от станции S до станции T вообще нельзя добраться.

Варианты считаются различными, если существует станция, которая является концом нового тоннеля в одном из них, но не в другом. Каждый вариант рассматривается независимо от остальных.

Помогите Вано узнать, сколько существует вариантов выбрать пару станций, между которыми можно проложить новый тоннель так, чтобы от станции S до станции T можно было доехать за время не более K .

Формат входных данных

Первая строка содержит два целых числа N и M ($2 \leq N \leq 200\,000$, $1 \leq M \leq 200\,000$) — количество станций и количество тоннелей.

Вторая строка содержит четыре целых числа S , T , L и K ($1 \leq S < T \leq N$, $1 \leq L \leq 10^9$, $1 \leq K \leq 10^{15}$) — номера станций, между которыми ездит больше всего граждан, время проезда по новому тоннелю и предельное допустимое время поездки.

Далее следуют M строк. В i -й из них записаны три целых числа A_i , B_i и C_i ($1 \leq A_i < B_i \leq N$, $1 \leq C_i \leq 10^9$) — номера станций, соединённых i -м тоннелем, и время проезда по нему.

Гарантируется, что все пары (A_i, B_i) различны.

Формат выходных данных

Выведите одно целое число — количество подходящих вариантов.

Примеры

стандартный ввод	стандартный вывод
7 8 6 7 1 2 1 2 1 1 6 1 2 3 1 2 4 1 3 5 1 3 7 1 4 5 1 5 6 1	4
3 2 1 3 1 2 1 2 1 2 3 1	3
6 4 2 5 1000000000 1 1 2 1000000000 2 3 1000000000 2 4 1000000000 5 6 1000000000	0

Замечание

В первом примере подходит прокладка нового тоннеля между станциями 6 и 7, 5 и 7, 1 и 7, а также 3 и 6. Во втором примере от станции 1 до станции 3 можно доехать за время 2 и без нового тоннеля, поэтому подходит любой из трёх вариантов. В третьем примере от станции 2 до станции 5 без нового тоннеля добраться нельзя, а время проезда по новому тоннелю больше K , поэтому подходящих вариантов нет.

Задача L. Граф и граф

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

Вам даны два связанных неориентированных графа с одинаковым количеством вершин. В обоих графах в какой-то вершине находится фишка. В первом графе фишка изначально находится в вершине s_1 , во втором графе фишка изначально находится в вершине s_2 . Далее **бесконечное** количество раз повторяются следующая операция:

- Пусть сейчас в первом графе фишка находится в вершине v_1 , а во втором графе в вершине v_2 .
- Выбирается какая-то вершина u_1 , смежная с v_1 , в первом графе.
- Выбирается какая-то вершина u_2 , смежная с v_2 , во втором графе.
- Фишки перемещаются в выбранные вершины: в первом графе фишка перемещается из v_1 в u_1 , во втором графе из v_2 в u_2 .
- Стоимость такой операции равна $|u_1 - u_2|$.

Определите минимально возможную суммарную стоимость всех операций или сообщите, что это значение будет бесконечно большим.

Формат входных данных

Каждый тест состоит из нескольких наборов входных данных. Первая строка содержит одно целое число t ($1 \leq t \leq 500$) — количество наборов входных данных. Далее следует описание наборов входных данных.

Первая строка каждого набора входных данных содержит три целых числа n , s_1 и s_2 ($2 \leq n \leq 1000$, $1 \leq s_1, s_2 \leq n$) — количество вершин в каждом графе, номер вершины в первом графе, где изначально находится фишка, и номер вершины во втором графе, где изначально находится фишка.

Вторая строка каждого набора входных данных содержит одно целое число m_1 ($1 \leq m_1 \leq 1000$) — количество рёбер в первом графе.

i -я из следующих m_1 строк содержит два целых числа a_i и b_i ($1 \leq a_i, b_i \leq n$, $a_i \neq b_i$) — номера концов i -го ребра в первом графе.

Следующая строка каждого набора входных данных содержит одно целое число m_2 ($1 \leq m_2 \leq 1000$) — количество рёбер во втором графе.

j -я из следующих m_2 строк содержит два целых числа c_j и d_j ($1 \leq c_j, d_j \leq n$, $c_j \neq d_j$) — номера концов j -го ребра во втором графе.

Гарантируется, что сумма n , сумма m_1 и сумма m_2 по всем наборам входных данных не превосходят 1000.

Гарантируется, что оба графа являются связными.

Формат выходных данных

Для каждого набора входных данных выведите одно целое число — минимальную суммарную стоимость всех операций или -1 , если это значение будет бесконечно большим.

Пример

стандартный ввод	стандартный вывод
3	0
4 1 1	-1
4	7
1 2	
2 3	
3 4	
4 1	
4	
1 2	
2 3	
3 4	
4 1	
4 1 2	
4	
1 2	
2 3	
3 4	
4 1	
4	
1 2	
2 3	
3 4	
4 1	
7 7 2	
7	
1 6	
2 1	
3 2	
3 4	
5 1	
7 3	
7 5	
6	
5 1	
5 6	
5 7	
6 3	
7 2	
7 4	

Замечание

В первом наборе входных данных можно построить бесконечную последовательность переходов в вершины 2, 3, 4, 1, 2, 3, 4, 1, ..., по которой фишка может двигаться как в первом, так и во втором графе.

Во втором наборе входных данных можно доказать, что стоимость любой операции будет больше 0, поэтому суммарная стоимость всех операций будет бесконечно большой.