

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. Путь в графе

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

В неориентированном графе требуется найти длину минимального пути между двумя вершинами.

Формат входных данных

Первым на вход поступает число N — количество вершин в графе ($1 \leq N \leq 100$). Затем записана матрица смежности (0 обозначает отсутствие ребра, 1 — наличие ребра). Далее задаются номера двух вершин — начальной и конечной.

Формат выходных данных

Необходимо вывести длину пути в ребрах. Если пути нет, нужно вывести -1 .

Пример

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

Задача D. Числа

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

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

В первой строке выведите единственное число k — количество чисел в последовательности.

В следующих k строках выведите 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	6
4321	1234
	2234
	3234
	4323
	4322
	4321

Задача Е. «0-1» BFS

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

Вам дан неориентированный взвешенный граф G на n вершин и m ребер и вершина s . Известно, что вес любого ребра равен либо единице, либо нулю. 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 $1 \leq a_i, b_i \leq n, 0 \leq c \leq 1$ — 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 1 3 1 1 4 0 2 3 0 8 3 1 7 3 0 3 6 1 2 5 1 6 9 1	0 1 1 0 2 2 1 2 3 -1

Задача F. Эвакуация

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

Сначала вводятся два натуральных числа N , K ($1 \leq N \leq 100000$, $1 \leq K \leq N$) — 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 число M ($1 \leq M \leq 100000$) — 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

Выведите N чисел, разделенных пробелом — для каждого из бункеров минимальное время, необходимое чтобы добраться до выхода. Считайте, что время перемещения по одному туннелю равно 1.

Примеры

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

Задача G. 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 Н. 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 I. Заправки-2

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

В стране n городов, некоторые из которых соединены между собой дорогами. Для того, чтобы проехать по одной дороге, требуется один бак бензина. Помимо этого у вас есть канистра для бензина, куда входит столько же топлива, сколько входит в бензобак.

В каждом городе бак бензина имеет разную стоимость. Вам требуется добраться из первого города в n -й, потратив как можно меньше денег.

В каждом городе можно заправить бак, заправить бак и канистру или же перелить бензин из канистры в бак. Это позволяет экономить деньги, покупая бензин в тех городах, где он стоит дешевле, но канистры хватает только на одну заправку бака!

Формат входных данных

В первой строке вводится число n ($1 \leq n \leq 100$), в следующей строке идет n чисел, i -е из которых задает стоимость бензина в i -м городе (всё это целые числа из диапазона от 0 до 100). Затем идет число M – количество дорог в стране, далее идет описание самих дорог. Каждая дорога задается двумя числами – номерами городов, которые она соединяет. Все дороги двухсторонние (то есть по ним можно ездить как в одну, так и в другую сторону), между двумя городами всегда существует не более одной дороги, не существует дорог, ведущих из города в себя.

Формат выходных данных

Требуется вывести одно число – суммарную стоимость маршрута или -1, если добраться невозможно.

Пример

стандартный ввод	стандартный вывод
4 1 10 2 15 4 1 2 1 3 4 2 4 3	2

Задача J. 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 перевёрнутом 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...XXXX 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 К. Настя и неожиданный гость

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

Если девушка не идет к Денису, то Денис идет к девушке. Руководствуясь таким принципом, молодой человек вышел из дома, купил цветы и направился к Насте.

По пути от его дома до дома его возлюбленной есть дорога из n полос, перейти которую за один зеленый свет не всегда возможно. Предвидя это, добрый мэр распорядился поставить в некоторых точках этой дороги островки безопасности. Каждый островок расположен после какой-то полосы, а также в начале и в конце дороги. На них пешеходы могут отдохнуть, набраться сил и дожждаться зеленого света.

Денис подошел к левому краю дороги ровно в тот момент, когда загорелся зеленый свет. Мальчик знает, что светофор сначала горит g секунд зеленым, а потом r секунд красным, потом опять g секунд зеленым и так далее.

Формально, дорогу можно представить как отрезок $[0, n]$. Изначально Денис стоит в точке 0. Его задача - попасть в точку n за минимальное возможное время.

Он знает множество различных целых чисел $d_1, d_2, \dots, d_m$, где $0 \leq d_i \leq n$ — координаты точек, в которых расположены островки безопасности. Только в одной из этих точек мальчик может находиться в момент, когда горит красный свет.

К сожалению, из-за волнения Денис не всегда может контролировать себя, поэтому сейчас на его передвижения наложены некоторые ограничения:

- Он обязан двигаться всегда, пока горит зеленый, ведь сложно стоять, когда тебя ждет такая девушка. Денис за 1 секунду может изменить свое положение на ± 1 . При этом он должен всегда оставаться внутри отрезка $[0, n]$.
- Он может менять направление движения только на островках безопасности (ведь это безопасно). Это означает, что если за предыдущую секунду мальчик изменил свое положение на $+1$, то если он находится на островке безопасности, он может изменить свое положение на ± 1 , иначе он может изменить свое положение вновь только на $+1$. Аналогично, если за предыдущую секунду он изменил свое положение на -1 , на островке безопасности он может изменить свое положение на ± 1 , а в любой другой точке вновь только на -1 .
- В момент, когда загорается красный, мальчик должен оказаться на каком-то островке безопасности. Он сможет продолжить движение в любом направлении, когда вновь загорится зеленый.

Считается, что Денис перешел дорогу как только его координата станет равна n .

Эта задача оказалась не так проста, ведь возможно, таким способом даже невозможно перейти дорогу. Так как у Дениса все мысли о любви, он не справился решить ее и попросил нас помочь ему в этом. Найдите минимальное возможное время, за которое он сможет перейти дорогу по таким правилам, либо установите, что это сделать невозможно.

Формат входных данных

В первой строке находится два целых числа n и m ($1 \leq n \leq 10^6, 2 \leq m \leq \min(n + 1, 10^4)$) — ширина дороги и количество островков безопасности.

Во второй строке находится m различных целых чисел $d_1, d_2, \dots, d_m$ ($0 \leq d_i \leq n$) — точки, в которых расположены островки безопасности. Гарантируется, что среди них есть 0 и n .

В третьей строке находится два целых числа g, r ($1 \leq g, r \leq 1000$) — время, которое на светофоре горит зеленый свет и время, которое на светофоре горит красный свет.

Формат выходных данных

Выведите одно целое число — минимальное время, за которое Денис может перейти дорогу по всем правилам.

Если перейти дорогу по всем правилам невозможно выведите -1 .

Примеры

стандартный ввод	стандартный вывод
15 5 0 3 7 14 15 11 11	45
13 4 0 3 7 13 9 9	-1

Замечание

В первом тесте оптимальный маршрут такой:

- за первый зеленый свет пройти до 7 и вернуться на 3. В этом случае, мы сменим направление движения в точке 7, что разрешено, поскольку в этой точке есть островок безопасности. В конце мы окажемся в точке 3, где также есть островок безопасности. Следующие 11 секунд мы должны подождать красный свет.
- за второй зеленый свет пройти до 14. Снова подождать красный свет.
- за 1 секунду перейти в 15. В итоге Денис оказывается в конце дороги.

Всего получается 45 секунд.

Во втором тесте невозможно перейти дорогу по всем правилам.

Задача L. Лабиринт

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

В одном из уровней компьютерной игры вы попали в лабиринт, состоящий из n строк, каждая из которых содержит m клеток. Каждая клетка либо свободна, либо занята препятствием. Стартовая клетка находится в строке r и столбце c . За один шаг вы можете переместиться на одну клетку вверх, влево, вниз или вправо, если она не занята препятствием. Вы не можете перемещаться за границы лабиринта.

К сожалению, ваша клавиатура крайне близка к поломке, поэтому вы можете переместиться влево не более x раз и вправо не более y раз. При этом ограничений на перемещения вверх и вниз нет, поскольку клавиши, используемые для движения вверх и вниз, всё ещё в идеальном состоянии.

Теперь вы для каждой клетки поля решили установить, можно ли выбрать такую последовательность нажатий, которая приведёт вас из стартовой в эту клетку. Посчитайте, сколько клеток поля обладают таким свойством.

Формат входных данных

Первая строка содержит два целых числа n, m ($1 \leq n, m \leq 2000$) — количество строк и столбцов в лабиринте, соответственно.

Вторая строка содержит два целых числа r, c ($1 \leq r \leq n, 1 \leq c \leq m$) — номер строки и столбца, на пересечении которых расположена стартовая клетка.

Третья строка содержит два целых числа x, y ($0 \leq x, y \leq 10^9$) — максимальное количество перемещений влево и вправо, соответственно.

Следующие n строк содержат описание лабиринта. Каждая из этих строк имеет длину m и состоит только из символов '.' и '*'. В i -й строке j -й символ соответствует клетке лабиринта с номерами строки и столбца i и j , соответственно. Символ '.' соответствует свободной клетке лабиринта, а символ '*' — клетке с препятствием.

Гарантируется, что стартовая клетка не занята препятствием.

Формат выходных данных

Выведите одно число — количество клеток лабиринта, достижимых из стартовой, включая её саму.

Примеры

стандартный ввод	стандартный вывод
4 5 3 2 1 2***. ...** *.....	10
5 5 5 4 3 1 **... **.*. ...*. .***.	16

Задача М. 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.

Задача N. Цепная реакция

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	512 мегабайт

Мало кто знает, почему магическая технология такая эффективная. На самом деле любой магический кристалл можно представить в виде n узлов и m двухсторонних связей между ними. Иногда в некоторых узлах вспыхивают искры энергии, которые затем распространяются в соседние узлы, из них — в соседние с ними, и так далее.

Недавно исследователи академии Пилтовера во главе с Хеймердингером выяснили, что делает магические кристаллы особенно сильными. Некоторые из связей между узлами являются *накопительными*. Это означает, что энергия может проходить через них только в определенные периоды времени, а вне этих периодов скапливается в концах этих связей и сохраняется в них.

Для каждой связи известно, сколько времени уходит на перемещение энергии из одного ее конца в другой. Также, для всех *накопительных* связей известно, что они пропускают энергию в одни и те же интервалы времени. Эти интервалы задаются списком пар моментов времени (x_i, y_i) , означающими, что между моментами времени x_i и y_i включительно каждая связь открыта и пропускает энергию. В остальные моменты времени они закрыты, и энергия «задерживается» в их концах.

Важно отметить, что если *накопительная* связь закрывается в тот момент, когда энергия перемещается по ней, энергия продолжает двигаться в ту сторону, в которую двигалась. Иными словами, если в момент t связь длиной w открыта, и в этот же момент t к одному ее концу приходит искра энергии, то в момент $t + w$ она доберется до второго конца, даже если связь закроется к этому времени.

Для создания нового магического оружия необходимо определить, в какой момент времени энергия впервые дойдет до узла v , если искра зародится в узле u в момент времени t_0 , или что энергия не дойдет до узла v вообще.

Формат входных данных

В первой строке ввода даны три целых числа n, m, k — количество узлов кристалла и связей между ними, а также количество пар моментов времени, между которыми все накопительные связи открыты ($2 \leq n \leq 10^5$; $1 \leq m \leq 10^6$, $1 \leq k \leq 10^5$).

В i -й из следующих m строк через пробел даны четыре целых числа a_i, b_i, w_i и f_i . Первые два числа a_i и b_i — номера узлов, которые соединяет i -я связь ($1 \leq a_i, b_i \leq n$; $a_i \neq b_i$). Число w_i — время, за которое энергия перемещается между концами связи ($1 \leq w_i \leq 10^9$). Число f_i задает, является ли i -я связь *накопительной*. Он равен 1, если является, и 0 иначе.

В каждой из следующих k строк записаны по два целых числа x_i, y_i — моменты времени, между которыми (включительно) каждая накопительная связь открыта и может перемещать энергию ($1 \leq x_i \leq y_i \leq 10^{18}$; $y_{i-1} < x_i$ для всех i).

В последней строке ввода даны три целых числа u, v и t_0 — номера стартового и конечного узла, а также время зарождения искры энергии в стартовом узле ($1 \leq u, v \leq n$; $1 \leq t_0 \leq 10^{18}$).

Формат выходных данных

Выведите одно целое число — через какое минимальное время энергия впервые дойдет от узла u до узла v , начав движение в момент времени t_0 . Если она не дойдет до v никогда, выведите «-1» (без кавычек).

Примеры

стандартный ввод	стандартный вывод
2 1 1 1 2 1 0 2 3 1 2 3	1
6 8 2 1 2 1 0 3 5 5 0 3 4 1 1 2 4 2 0 3 6 3 1 4 6 1 0 5 6 1 1 1 6 1 1 6 6 39 40 2 5 8	32