Задача A. 2-SAT

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

Формулировка `2-SAT`: нужно подобрать значения n булевых переменных так, чтобы все m утверждений вида $(x_{i_1} = e_1) \vee (x_{i_2} = e_2)$ обратились в истину. В данной задаче вам гарантируется, что решение существует.

Формат входных данных

Входной файл состоит из одного или нескольких тестов.

Каждый тест описывается следующим образом. На первой строке число переменных n и число утверждений m. Каждая из следующих m строк содержит числа i_1, e_1, i_2, e_2, задает утверждение $(x_{i_1} = e_1) \vee (x_{i_2} = e_2)$ ($0 \leqslant i_j < n$, $0 \leqslant e_j \leqslant 1$). Ограничения: сумма всех n не больше 100 000, сумма всех m не больше 300 000.

Формат выходных данных

Для каждого теста выведите строку из n нулей и единиц — значения переменных. Если у данной задачи `2-SAT` есть несколько решений, выведите любое.

Пример

стандартный ввод	стандартный вывод
1 0 2 2 0 0 1 0 0 1 1 1 3 4 0 1 1 0 0 0 2 1 1 1 2 0 0 0 0 1	0 01 000

Задача B. Интересная экскурсия

Имя входного файла:	`стандартный ввод`
Имя выходного файла:	`стандартный вывод`
Ограничение по времени:	4 секунды
Ограничение по памяти:	512 мегабайт

Во Флатландии n городов, соединенных m односторонними дорогами.

Туристическая компания планирует разработать живописный циклический маршрут по дорогам Флатландии. Маршрут должен начинаться и заканчиваться в одном и том же городе и проходить по дорогам в их направлении, посещая промежуточные города. Маршрут может посещать один город несколько раз, но должен проходить по каждой дороге не более одного раза.

Каждая дорога характеризуется типом своего пейзажа, который задается числом от 1 до m. Чтобы туристический маршрут был живописным, любые две соседние дороги в этом маршруте должны иметь разный тип пейзажа. Это же требование относится к первой и последней дороге маршрута, чтобы можно было начинать путешествовать, начиная с любого города маршрута.

Помогите компании разработать удовлетворяющий этим критериям маршрут, либо выясните, что сделать это невозможно.

Формат входных данных

Входные данные состоят из нескольких тестов. В первой строке находится число T — количество тестов ($1 \leqslant T \leqslant 10^5$).

Первая строка описания каждого теста содержит два целых числа n и m — количество городов и дорог ($2 \leqslant n, m \leqslant 2 \cdot 10^5$). Следующие m строк содержат по три целых числа и описывают дороги в формате u_i v_i c_i — город, из которого выходит i-я дорога, город, в который она ведет, и тип её пейзажа ($1 \leqslant u_i, v_i \leqslant n$; $1 \leqslant c_i \leqslant m$; $u_i \neq v_i$).

Сумма n по всем тестам не превосходит $2 \cdot 10^5$. Сумма m по всем тестам не превосходит $2 \cdot 10^5$.

Формат выходных данных

Выведите ответ для каждого теста.

Если искомого маршрута не существует, следует вывести число «-1». Иначе, выведите число k — длину маршрута ($2 \leqslant k \leqslant m$). В следующей строке выведите k чисел $e_1, e_2, \ldots, e_k$ — номера дорог в том порядке, в каком они идут в этом маршруте. Все номера e_i должны быть различны. Если подходящих маршрутов несколько, можно вывести любой из них.

Пример

стандартный ввод	стандартный вывод
3 5 8 1 4 1 2 4 1 4 5 2 3 2 2 5 3 1 3 2 3 5 2 2 2 1 3 4 5 1 2 2 2 3 1 2 4 4 4 1 2 3 1 2 2 3 1 2 1 1 2 2 2 1 1	4 3 5 6 2 -1 2 2 3

Задача C. Эйлеров путь

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

Дан неориентированный связный граф, не более трех вершин имеет нечетную степень. Требуется определить, существует ли в нем путь, проходящий по всем ребрам.

Если такой путь существует, необходимо его вывести.

Формат входных данных

Первая строка входного файла содержит натуральное число n — количество вершин графа ($1 \leqslant n \leqslant 100\,000$). Далее следуют n строк, задающих ребра. В i-й из этих строк находится число m_i — количество ребер, инцидентных вершине i. Далее следуют m_i натуральных чисел — номера вершин, в которые ведут ребро из i-й вершины.

Граф может содержать кратные ребра, но не содержит петель.

Граф содержит не более 300 000 ребер.

Формат выходных данных

Если решение существует, то в первую строку выходного файла выведите одно число k — количество ребер в искомом маршруте, а во вторую $k+1$ число — номера вершин в порядке их посещения.

Если решений нет, выведите в выходной файл одно число `-1`.

Если решений несколько, выведите любое.

Пример

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

Задача D. Xor the Graph

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

Дан неориентированный граф, состоящий из N вершин и M рёбер. Каждому ребру присвоено значение, 0 или 1. Операция состоит в том, чтобы выбрать путь (не обязательно простой), и изменить значения, записанные на каждом его ребре на противоположные. Сделайте значения на всех рёбрах нулями за минимальное число операций.

Формат входных данных

На первой строке заданы два целых числа N и M ($1 \leqslant N \leqslant 10^5$, $0 \leqslant M \leqslant 10^5$).

В каждой из следующих M строк записаны по три целых числа a, b и c. Они означают, что существует ребро между вершинами (a, b), изначальное значение которого равно c ($1 \leqslant a, b \leqslant n$, $0 \leqslant c \leqslant 1$).

Формат выходных данных

Выведите количество операций K на первой строке.

В последующих K строках выведите пути, которые вы выбрали:

- Первым числом выведите P — количество вершин в пути
- Затем выведите P целых чисел $v_1, v_2, \ldots, v_P$ — индексы вершин. Для всех $1 \leqslant i \leqslant P - 1$ (v_i, v_{i+1}) должно являться ребром в заданном графе.

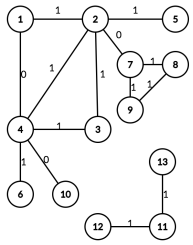
Решение будет считаться верным, если оно использует минимальное количество операций, и при этом $\sum P \leqslant 4 * M$.

Пример

стандартный ввод	стандартный вывод
13 14 1 2 1 2 3 1 3 4 1 2 4 1 1 4 0 4 6 1 4 10 0 2 5 1 2 7 0 7 8 1 8 9 1 9 7 1 11 12 1 11 13 1	3 5 1 2 3 4 6 8 4 2 7 8 9 7 2 5 3 12 11 13

Замечание

Ниже приведён граф в первом тесте.



Задача E. Сбалансированная строка

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

Надо найти лексикографически минимальную строку s длины n, состоящую из 0 и 1, что она удовлетворяет m ограничениям: на отрезке $[L_i, R_i]$ должно быть равное количество нулей и единиц. Гарантируется, что длина каждого отрезка чётная.

Формат входных данных

Первая строка содержит два числа n и m ($2 \leqslant n \leqslant 10^6, 1 \leqslant m \leqslant 2 \cdot 10^5$) — длина строки и количество ограничений соответственно.

Следующие m строк содержат отрезки L_i и R_i ($1 \leqslant L_i < R_i \leqslant n, R_i - L_i + 1 \equiv 0 \bmod 2$).

Формат выходных данных

Выведите лексикографически минимальную строку, удовлетворяющую этим требованиям.

Примеры

стандартный ввод	стандартный вывод
4 2 1 2 3 4	0101
6 2 1 4 3 6	001100
20 10 6 17 2 3 14 19 5 14 10 15 7 20 10 19 3 20 6 9 7 12	00100100101101001011

Задача F. Красно-синий граф

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

Вам дан граф на n вершинах с $\frac{n \cdot (n-1)}{2}$ ориентированными ребрами. Ребро между вершинами (u, v) существует, если $u < v$. Каждое ребро покрашено либо в синий, либо в красный цвет. Проверьте, существует ли такая пара вершин (v, u), что существует:

- путь $v \to u$, состоящий только из красных ребер
- путь $v \to u$, состоящий только из синих ребер

Формат входных данных

В первой строке вводится число n ($1 \leqslant n \leqslant 5000$). В следующих $n-1$ строках вводится описание графа.

i-я строка описания графа будет содержать строку из $n - i$ символов 'R' или 'B'. j-й символ обозначает цвет ребра $i \to (i + j)$.

Формат выходных данных

Если искомая пара v, u существует, выведите «YES». Иначе выведите «NO»

Примеры

стандартный ввод	стандартный вывод
3 RB R	YES
3 RR R	NO

Задача G. Проездной

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

JOI-кун живёт в городе с n станциями, занумерованными от 1 до n. Есть m железных дорог, пронумерованных от 1 до m, и i-я дорога соединяет станции a_i и b_i в обоих направлениях, притом стоимость проезда равна c_i.

JOI-кун живёт рядом со станцией s и ходит в старшую школу IOI рядом со станцией t. Он планирует купить проездной между этими станциями. Когда он покупает проездной, он должен выбрать какой-то кратчайший путь между s и t. Используя проездной, он может использовать любую железную дорогу на выбранном пути в обоих направлениях, не платя за проезд.

JOI-кун часто ездит в книжные маганизы рядом со станциями u и v. Поэтому он хочет купить проездной так, чтобы минимальная стоимость проезда от u до v была минимальна.

Когда он перемещается из станции u на станцию v, он сперва выбирает путь от станции u до станции v. Тогда за дороги в пути, входящие в проездной, он заплатит 0 иен, а за не входящие в проездной — стоимость проезда через них, то есть c_i для дороги i.

Найдите минимально возможную стоимость пути из u в v, если JOI-кун выберет путь для проездного оптимально.

Формат входных данных

Первая строка содержит два целых числа n и m ($2 \leqslant n \leqslant 10^5$, $1 \leqslant m \leqslant 2 \cdot 10^5$).

Вторая строка содержит два целых числа s и t ($1 \leqslant s, t \leqslant n$, $s \neq t$).

Третья строка содержит два целых числа u и v ($1 \leqslant u, v \leqslant n$, $u \neq v$, $s \neq u$ или $t \neq v$).

Далее следуют m строк, описывающие железные дороги, i-я из них содержит три целых числа a_i, b_i, c_i ($1 \leqslant a_i < b_i \leqslant n$, $1 \leqslant c_i \leqslant 10^9$). Для всех $1 \leqslant i < j \leqslant m$ верно $a_i \neq a_j$ или $b_i \neq b_j$.

Гарантируется, что JOI-кун может добраться от любой станции до любой другой, перемещаясь только по железным дорогам.

Формат выходных данных

Выведите одно число — ответ.

Примеры

стандартный ввод	стандартный вывод
6 6 1 6 1 4 1 2 1 2 3 1 3 5 1 2 4 3 4 5 2 5 6 1	2
6 5 1 2 3 6 1 2 1000000000 2 3 1000000000 3 4 1000000000 4 5 1000000000 5 6 1000000000	3000000000
8 8 5 7 6 8 1 2 2 2 3 3 3 4 4 1 4 1 1 5 5 2 6 6 3 7 7 4 8 8	15
5 5 1 5 2 3 1 2 1 2 3 10 2 4 10 3 5 10 4 5 10	0

Задача H. Диаграммы Юнга выходят в интернет

Имя входного файла:	`стандартный ввод`
Имя выходного файла:	`стандартный вывод`
Ограничение по времени:	3 секунды
Ограничение по памяти:	256 мегабайт

Рома и Сеня дали контест из 10 задач про диаграмму Юнга и теперь вынуждены скрываться от ДемидКомНадзора. Но они слишком любят диаграммы Юнга и хотят делиться новыми открытиями. Однако держаться вместе очень рискованно, так как в случае чего повяжут их обоих, поэтому они вынуждены общаться через Интернет. Но "обычный Интернет" полностью контролируется ДемидКомНадзором, поэтому они пользуются даркнетом.

В даркнете любое сообщение может проделать длинный запутанный путь до получателя через множество серверов. Более того, оно даже может проходить через один и тот же сервер несколько раз. За счет этого сообщение сложнее отследить.

Компьютер Ромы связан с сервером 1, а компьютер Сени — с сервером n.

ДемидКомНадзор хочет перехватить Ромино сообщение с диаграммой Юнга. Для этого ему необходимо взломать такой сервер, что сообщение, посланное Ромой, по любому пути к Сене пройдет через этот сервер **ровно один** раз.

Найдите все подходящие сервера.

Формат входных данных

В первой строке файла дано количество тестовых примеров t ($1 \leqslant t \leqslant 500$).

Каждый тестовый пример выглядит так: в первой строке даны два числа: n и m ($2 \leqslant n \leqslant 5 \cdot 10^5$, $0 \leqslant m \leqslant 10^6$), число серверов и число прямых соединений между серверами.

В каждой из последующих m строк содержится упорядоченная пара чисел a и b ($1 \leqslant a, b \leqslant n$), это означает, что с сервера a можно переслать сообщение напрямую на сервер b.

Гарантируется, что эти упорядоченные пары не повторяются внутри одного тестового примера.

Так же гарантируется, что и сумма по n, и сумма по m по всем тестовым примерам не превосходит 10^6.

Формат выходных данных

Для каждого тестового примера требуется вывести две строки: в первой число подходящих серверов, а во второй — номера этих серверов в порядке их следования на пути от a до b через пробел.

Пример

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

Задача I. Подземная система

Имя входного файла:	`стандартный ввод`
Имя выходного файла:	`стандартный вывод`
Ограничение по времени:	3 секунды
Ограничение по памяти:	256 мегабайт

Подземная система метро night-city состоит из N станций и M железнодорожных линий. Станции пронумерованы от 1 до N. Каждой линией управляет какая-то компания с присвоенным номером.

Железнодорожная линия с номером i соединяет станции p_i и q_i двусторонним путем, которым оперирует компания с номером c_i.

Вы можете пересаживаться с линий, которые оперируются одной компанией, на линии, которые оперируются другой компанией за 1 бурль. Тем не менее, если пассажир только что проехал по пути, которым управляет компания с номером i он может бесплатно проехать по пути, управляемым такой же компанией.

К примеру, если пассажир проедет по ребрам **2, 3, 3, 1, 2, 2, 2, 3**, где числа — номера компаний, то стоимость такой поездке составит 5 бурлей (1 бурль, чтобы сесть на поезд и по 1 бурлю за изменение используемых компаний)

Вы хотите узнать минимальную стоимость проезда от станции с номером 1 до станции с номером n.

Формат входных данных

В первой строке содержатся два целых числа N, M ($2 \leqslant N \leqslant 10^5$, $0 \leqslant M \leqslant 2 \cdot 10^5$) — количество станций и количество железнодорожных линий соответственно.

В следующих m строках содержатся по три целых числа p_i, q_i, c_i ($1 \leqslant p_i, q_i \leqslant N$, $1 \leqslant c_i \leqslant 10^6$) — вершины, соединяемые i-м ребром и компания, которая управляет этим путем.

Формат выходных данных

В единственной строке выходных данных выведите одно целое число — минимальную стоимость проезда из вершины 1 до вершины n. Если попасть на станцию с номером n невозможно, выведите -1.

Примеры

стандартный ввод	стандартный вывод
3 3 1 2 1 2 3 1 3 1 2	1
8 11 1 3 1 1 4 2 2 3 1 2 5 1 3 4 3 3 6 3 3 7 3 4 8 4 5 6 1 6 7 5 7 8 5	2
2 0	-1

Задача J. Теория чисел

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

Найдите минимальную возможную сумму цифр в десятичной записи числа такого, которое делится на K.

Формат входных данных

В первой строке входных данных содержится целое число K ($2 \leqslant K \leqslant 10^5$)

Формат выходных данных

В единственной строке выходных данных выведите ответ на задачу.

Примеры

стандартный ввод	стандартный вывод
6	3
41	5
79992	36

Задача K. Дизайнерский лифт

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

Дизайн-студия Артемия Индюкова получила заказ на разработку очень пафосного лифта для нового небоскреба. За работу взялся сам Артемий, отличающийся редкой неадекватностью. У него есть идея-фикс: для управления лифтом достаточно четырех кнопок. Кнопки должны быть следующие:

- Поднятся на A этажей вверх
- Поднятся на B этажей вверх
- Поднятся на C этажей вверх
- Спустится на первый этаж

Изначально лифт находится на первом этаже. Пассажир лифта использует первые три кнопки, чтобы попасть на тот этаж, на который он хочет. Если пассажир пытается подняться вверх на A, B или C этажей, а такого этажа в здании не существует (т.е. пассажир хочет подняться выше N-го, последнего этажа), то лифт никуда не едет.

Заказчики проекта оказались с юмором и вместе с отказом от футуристичного дизайна решили оценить адекватность Артемия по шкале от 1 до N. Оценка адеватности равна количеству этажей, на которые можно попасть с первого с помощью такого лифта. Помогите им в этом.

Формат входных данных

Первая строка содержит число N — высоту небоскреба ($1 \leqslant N \leqslant 10^{18}$).

Вторая строка содержит три числа A, B и C, задающие параметры кнопок ($1 \leqslant A, B, C \leqslant 100000$).

Формат выходных данных

Выведите единственное число — оценку адекватности Артемия Индюкова.

Пример

стандартный ввод	стандартный вывод
15 4 7 9	9

Задача L. Испытание автомобиля на льду

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

Вам поручено провести испытание нового автомобиля, способного хорошо ездить по льду. Чтобы как можно больше впечатлить начальство, вы решили провести испытание на максимальной возможной скорости.

На ледяном озере, на котором будет проходить испытание, условно отмечено n точек, пронумерованных целыми числами от 1 до n. Между некоторыми парами точек можно проехать напрямую. При этом для каждой такой пары известна максимальная скорость, которую может развить автомобиль, проезжая между этими точками. Если автомобиль превысит эту скорость, то он провалится под лёд, что недопустимо. Вы хотите, чтобы автомобиль проехал по некоторому пути от точки 1 до точки n — либо напрямую, либо через несколько промежуточных точек (в этом случае между каждыми двумя соседними точками маршрута должно быть возможно напрямую переместиться, включая переправу из точки 1 в первую промежуточную точку и переправу из последней промежуточной точки в точку n).

Задачу осложняет тот факт, что испытание проходит на льду. Чтобы предотвратить скольжение, конструкторы установили шипы на колёса автомобиля. В пути они немного стираются, что не было учтено при проектировании спидометра автомобиля. По этой причине его тонкая настройка сбивается, и реальная скорость автомобиля оказывается выше, чем фиксируемая приборами. К счастью, ваши инженеры вовремя спохватились и определили, что разница между скоростью и показаниями приборов на каждом прямом отрезке между двумя точками равна некоторому числу a_k, зависящему только от k — количества уже пройденных прямых отрезков (включая тот, который автомобиль проходит сейчас). Другими словами, если автомобиль движется по маршруту $i_1 \to i_2 \to i_3 \to \ldots \to i_\ell$, где $i_1, i_2, \ldots, i_\ell$ — номера последовательно посещаемых точек, то на отрезке $i_k \to i_{k+1}$ скорость автомобиля будет превосходить скорость, показываемую приборами, ровно на величину a_k. Инженеры смогли узнать числа $a_1, a_2, \ldots, a_{n-1}$. Интересно, что эта последовательность оказалась неотрицательной и неубывающей, то есть $0 \leqslant a_1 \leqslant a_2 \leqslant \ldots \leqslant a_{n-1}$. Причина этого в том, что при всё большем стирании шипов калибровка спидометра всё больше сбивается, из-за чего его показания всё сильнее расходятся с реальной скоростью автомобиля. Обратите внимание, что если $a_1 > 0$, то приборы *всегда* показывают число, строго меньшее скорости автомобиля, даже на самом первом отрезке пути.

Вы должны найти такой путь между точкой 1 и точкой n, что между каждой парой соседних точек вдоль пути автомобиль сможет проехать напрямую, не провалившись под лёд. Найденный путь должен быть простым, то есть никакая точка не должна встречаться в пути более одного раза. Первая точка пути должна иметь номер 1, последняя — номер n. Среди всех путей, подходящих под эти ограничения, вы должны найти тот, в котором минимальная скорость, **показываемая приборами** за весь путь, как можно больше. Вдоль каждого отрезка пути автомобиль должен двигаться с постоянной целой положительной скоростью (на разных отрезках можно перемещаться с разной скоростью), при этом скорость на спидометре также всегда должна быть положительной.

Формат входных данных

В первой строке находится два целых числа n и m — количество точек на озере и количество пар точек, между которыми автомобиль может проехать напрямую ($2 \leqslant n \leqslant 200\,000$, $0 \leqslant m \leqslant 200\,000$). Во второй строке находится $n-1$ целых чисел $a_1, a_2, \ldots, a_{n-1}$ ($0 \leqslant a_1 \leqslant a_2 \leqslant \ldots \leqslant a_{n-1} \leqslant 10^9$). В каждой из следующих m строк находится по три целых числа s_i, f_i, v_i — номера двух точек, между которыми автомобиль может проехать напрямую (в любую сторону), и максимальная реальная скорость, которую может развить автомобиль, проезжая между этой парой точек ($1 \leqslant s_i, f_i \leqslant n$, $s_i \neq f_i$, $1 \leqslant v_i \leqslant 10^9$). Гарантируется, что все m пар точек различны.

Формат выходных данных

Если ни одного подходящего пути от точки 1 до точки n не существует, то выведите единственное

целое число -1.

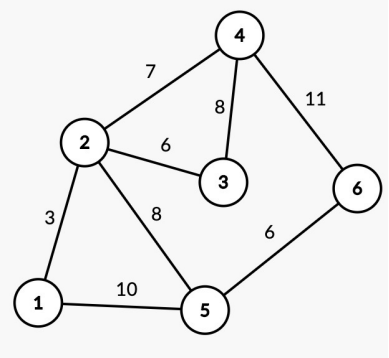
Иначе в первой строке выведите единственное целое число s — минимальную скорость автомобиля по приборам, которая будет при проезде вдоль вашего пути ($1 \leqslant s \leqslant 10^9$). Во второй строке выведите единственное целое число ℓ — количество точек в найденном пути ($2 \leqslant \ell \leqslant n$). В третьей строке выведите ℓ целых чисел $p_1, p_2, \ldots, p_\ell$, разделённых пробелами — номера точек в том порядке, в котором они следуют вдоль вашего пути. Путь должен начинаться в точке 1 и заканчиваться в точке n, поэтому должно быть верно $p_1 = 1$ и $p_\ell = n$. Все p_i должны быть попарно различными и удовлетворять ограничениям $1 \leqslant p_i \leqslant n$. Наконец, для всех целых $1 \leqslant i \leqslant \ell - 1$ между точками p_i и p_{i+1} должно быть возможно проехать напрямую. Если существует несколько путей, в которых s максимально возможное, выведите любой.

Примеры

стандартный ввод	*стандартный вывод*
6 8 0 3 3 4 8 1 2 3 3 2 6 2 4 7 2 5 8 5 1 10 6 5 6 4 6 11 3 4 8	4 5 1 5 2 4 6
4 2 1 2 2 1 3 100 4 1 1	-1
3 3 0 4 1 2 4 1 3 3 2 3 9	4 3 1 2 3

Пояснения к примерам

Вот картинка, иллюстрирующая первый тест:



Максимальная скорость по приборам, которую может развить автомобиль, проезжая между первой и пятой точкой, равна 10, между пятой и второй равна 5, между второй и четвёртой равна 4, а между четвёртой и шестой равна 7. Можно доказать, что скорость по приборам, превосходящую 4, нельзя развить ни на каком пути из первой точки в шестую.

Во втором тесте ни одного подходящего пути не существует, потому что скорость автомобиля по приборам всегда должна быть положительной.

Задача M. Странствующий торговец

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

Вы прибыли в Австралию, где есть n рынков, соединённых m односторонними дорогами, путешествие по каждой дороге занимает определённое количество минут.

На рынках торгуются k предметами. Каждый предмет имеет определённую стоимость покупки или продажи. Бывает так, что на рынке можно только купить товар или только продать товар, а также бывает, что рынку вообще не интересен товар. Вы можете считать, что если на рынке есть товар, его есть бесконечно много, а также, если рынок готов покупать товар, он готов покупать бесконечно много.

Чтобы как можно быстрее заработать денег вы хотите найти самый эффективный цикл. Цикл — это путь, который начинается в каком-то рынке v с пустым рюкзаком, проходит по дорогам и рынкам (возможно, по пути покупаются и продаются товары), и возвращается в вершину v, опять с пустым рюкзаком. Цикл может посещать дорогу или рынок несколько раз. Когда вы покупаете товар, вы кладёте его в рюкзак. Однако в рюкзак можно положить **не более одного товара**. Вы можете считать, что независимо от того, сколько у вас денег, вы можете купить товар.

Выгода цикла — это суммарное количество денег, которое вы заработали на продажах, минус количество денег, которые вы потратили на покупку. Длительность цикла — количество минут, которые вы потратите, чтобы пройти его. Эффективность цикла — отношение его выгоды к длительности.

Найдите максимальную эффективность среди всех циклов со строго положительной длительностью. Вы должны найти это значение, округленное вниз. Если такого цикла не существует, ответ равен 0.

Формат входных данных

Первая строка содержит три целых числа n, m, k ($1 \leqslant n \leqslant 100$, $1 \leqslant m \leqslant 9900$, $1 \leqslant k \leqslant 1000$).

Затем следуют n строк, i-я из которых содержит $2k$ чисел $b_{i,1}, s_{i,1}, b_{i,2}, s_{i,2}, \ldots, b_{i,k}, s_{i,k}$ ($0 < s_{i,j} \leqslant b_{i,j} \leqslant 10^9$). Для всех $1 \leqslant j \leqslant k$ пара чисел $b_{i,j}$ и $s_{i,j}$ означает цену, по которой вы можете купить и продать товар j на i-м рынке, соответственно. Если товар не может быть куплен или продан, тогда значение равно -1.

Далее следуют m строк, p-я из которых содержит три целых числа v_p, w_p и t_p ($v_p \neq w_p$, $1 \leqslant t_p \leqslant 10^7$), описывающих дорогу из v_p в w_p, которая занимает t_p минут.

Гарантируется, что не существует такой пары рёбер $1 \leqslant p < q \leqslant m$, что $(v_p, w_p) = (v_q, w_q)$.

Формат выходных данных

Выведите одно число — ответ.

Пример

стандартный ввод	стандартный вывод
4 5 2 10 9 5 2 6 4 20 15 9 7 10 9 -1 -1 16 11 1 2 3 2 3 3 1 4 1 4 3 1 3 1 1	2