

Задача А. 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 В. 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 С. Пифагоров экспресс (для C++)

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2.5 секунд
Ограничение по памяти: 256 мегабайт

В стране Флатландии есть n городов, расположенных в целочисленных точках плоскости. Транспортная система Флатландии настолько развита, что между любыми двумя городами ходит экспресс имени Пифагора. С помощью него можно добраться от города с координатами x_1, y_1 до города с координатами x_2, y_2 за время $(x_1 - x_2)^2 + (y_1 - y_2)^2$. По каждой линии ходит достаточно поездов, и временем на пересадки можно пренебречь.

Сообщение с внешним миром во Флатландии продумано несколько хуже, и единственный аэропорт международного сообщения находится в городе с номером s . Вам же хочется попасть в город с номером t . Определите, за какое наименьшее время это можно сделать.

Формат входных данных

В первой строке задано число городов n ($1 \leq n \leq 2 \cdot 10^4$). В следующих n строках заданы координаты городов x_i, y_i ($|x_i|, |y_i| \leq 10^4$). В последней строке даны s и t — номера начального и конечного города в пути ($1 \leq s, t \leq n$).

Формат выходных данных

Выведите одно число — минимальное время, за которое можно добраться из s в t .

Пример

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

Задача D. 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 магазины 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 ($2 \leq n \leq 10^5$, $1 \leq m \leq 2 \cdot 10^5$).

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

Третья строка содержит два целых числа u и v ($1 \leq u, v \leq n$, $u \neq v$, $s \neq u$ 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 ($1 \leq a_i < b_i \leq n$, $1 \leq c_i \leq 10^9$). Для всех $1 \leq i < j \leq m$ 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 Е. Дед и мопед

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мегабайт

Дед Максим собирается в путешествие по Флатландии. К сожалению, из доступных средств передвижения у него есть только мопед, запас хода которого ограничен. Более точно, если бак мопеда полностью заполнен, то мопед может проехать не более s километров без дополнительной дозаправки.

Всего во Флатландии есть n городов, пронумерованных от 1 до n . В некоторых городах находятся заправки, и если в городе есть заправка, то в этом городе дед Максим может полностью наполнить бак. К сожалению, заправки присутствуют лишь в k городах. Также во Флатландии есть m дорог, i -я из которых соединяет города u_i и v_i и имеет длину c_i километров. По каждой дороге можно перемещаться в обоих направлениях.

Дед Максим начинает свое путешествие в городе с номером 1 с полным баком (в городе 1 есть заправка). Помогите ему определить, до каких городов он сможет добраться на мопеде.

Формат входных данных

В первой строке записаны четыре целых числа n, m, k, s ($1 \leq n \leq 150\,000$, $0 \leq m \leq 150\,000$, $1 \leq k \leq n$, $1 \leq s \leq 10^9$) — количество городов, количество дорог, количество заправок и объем бака.

В следующих m строках записаны по три целых числа u_i, v_i, c_i ($1 \leq u_i, v_i \leq n$, $u_i \neq v_i$, $1 \leq c_i \leq s$) — начало, конец и длина i -й дороги. Гарантируется, что не существует двух дорог, соединяющих одинаковую пару городов.

В следующей строке записаны k целых чисел p_i ($1 \leq p_i \leq n$) — номера городов с заправками. Гарантируется, что заправка присутствует в городе 1.

Формат выходных данных

В первой строке выведите единственное число x : количество гордов, до которых дед Максим может добраться.

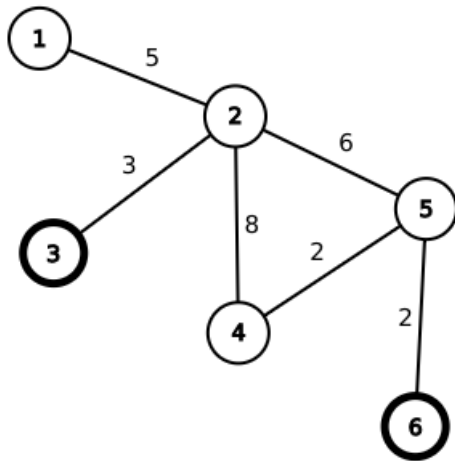
Во второй строке выведите x целых чисел — номера подходящих городов в порядке возрастания.

Пример

стандартный ввод	стандартный вывод
6 6 3 10 1 2 5 2 3 3 2 4 8 2 5 6 4 5 2 5 6 2 1 3 6	4 1 2 3 5

Замечание

Рисунок ниже иллюстрирует первый пример из условия:



Из начального города с номером 1 можно доехать до городов с номерами 2 и 3 без дополнительных дозаправок. Также можно доехать до города с номером 3, пополнить там бак и доехать до города с номером 5. До городов 4 и 6 добраться невозможно, так как минимальное расстояние от достижимой заправки до них равно 11.

Задача F. Очередная задача на Дейкстру

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

Обратите внимание на влияние различных подходов к реализации на константу решения.

Саша снова решил задачу на сборах и дал её вам. Теперь решите её и вы.

Вы приземляетесь на массивной решетке для винограда с общим количеством 2^k лилий и m виноградных лоз. Лилии пронумерованы от 0 до $2^k - 1$, а i -я виноградная лоза соединяет лилии с номерами x_i и y_i .

Вы можете потратить c_i минут, чтобы пройти по i -й виноградной лозе, что означает, что вы можете перемещаться от x_i к y_i или наоборот. В качестве альтернативы вы можете потратить a_k минут, чтобы телепортироваться от x к y , где x и y — любые две лилии, а k — количество различающихся битов в их двоичных представлениях. Например, двоичное представление числа 3 — это 011, а двоичное представление числа 5 — это 101. Они различаются в двух битах, поэтому количество минут, необходимых для телепортации от 3 к 5, равно a_2 .

Предположим, вы приземляетесь точно на лилию с номером s . Ваша задача — определить минимальное количество минут, необходимых для путешествия от s к каждой другой лилии.

Формат входных данных

Первая строка содержит три целых числа k , m и s ($1 \leq k \leq 17$; $1 \leq m \leq 2 \cdot 10^5$; $0 \leq s \leq 2^k - 1$).

Вторая строка содержит k целых чисел $a_1, a_2, \dots, a_k$ ($0 \leq a_i \leq 2^{30} - 1$).

Затем следуют m строк. i -я из этих строк содержит три целых числа x_i, y_i и c_i ($0 \leq x_i, y_i \leq 2^k - 1$; $0 \leq c_i \leq 2^{30} - 1$).

Формат выходных данных

Выведите одну строку, содержащую 2^k целых чисел D_i ($0 \leq i \leq 2^k - 1$), где D_i представляет собой минимальное количество минут, необходимых для путешествия от s к i .

Пример

стандартный ввод	стандартный вывод
3 6 2 17 14 11 0 2 3 4 2 9 2 2 1 2 2 6 7 0 5 4 2 9	3 14 0 17 9 11 17 8

Задача G. 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 ($1 \leq n \leq 100$, $1 \leq m \leq 9900$, $1 \leq k \leq 1000$).

Затем следуют n строк, i -я из которых содержит $2k$ чисел $b_{i,1}, s_{i,1}, b_{i,2}, s_{i,2}, \dots, b_{i,k}, s_{i,k}$ ($0 < s_{i,j} \leq b_{i,j} \leq 10^9$). Для всех $1 \leq j \leq k$ 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 ($v_p \neq w_p$, $1 \leq t_p \leq 10^7$), описывающих дорогу из v_p в w_p , которая занимает t_p минут.

Гарантируется, что не существует такой пары рёбер $1 \leq p < q \leq m$, 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

Задача Н. Геома или нет?

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

На плоскости расположены n точек, координаты y которых равны -9999 , 0 или 9999 . Пусть P — это множество этих n точек. Ваша задача — покрыть все точки в P минимальным возможным количеством квадратов: каждый квадрат имеет длину стороны $10\,000$ и стороны, параллельные координатным осям. Каждый квадрат покрывает все точки внутри и на границе квадрата.

Формат входных данных

Входные данные начинаются с строки, состоящей из одного целого числа n ($1 \leq n \leq 300\,000$), представляющего количество входных точек в P . В каждой из следующих n строк находятся два целых числа x и y , представляющие x - и y -координаты точки в P , соответственно ($-10^9 \leq x \leq 10^9$ и $y \in \{-9999, 0, 9999\}$). Вы можете предположить, что все n входные точки различны.

Формат выходных данных

Выведите ровно одну строку. Строка должна состоять из одного целого числа, представляющего минимально возможное количество t , такое что существуют t квадратов с длиной стороны $10\,000$ и сторонами, параллельными координатным осям, объединение которых покрывает все входные точки в P .

Примеры

стандартный ввод	стандартный вывод
5 0 9999 0 0 0 -9999 200 0 10000 9999	2
5 10 -9999 0 0 3 9999 9000 -9999 10003 9999	2
6 10 -9999 0 0 3 9999 9000 -9999 10003 -9999 10003 9999	3

Задача I. Наименьшее кратное

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

Дано число X и множество цифр D . Требуется дописать к X минимальное количество цифр из D , чтобы получившееся число делилось на k . При этом получившееся число должно быть минимально возможным.

Асимптотика $O(kD)$.

Формат входных данных

Первая строка входного файла содержит два натуральных числа X и k ($1 \leq X \leq 10^{1000}$, $2 \leq k \leq 10^5$). Во второй строке записано количество цифр во множестве D . В третьей строке через пробел записаны эти цифры.

Формат выходных данных

Единственная строка должна содержать минимальное число, полученное из X дописыванием цифр из D и кратное k . Если такого числа не существует, выведите -1 .

Пример

стандартный ввод	стандартный вывод
102 101 3 1 0 3	10201

Задача J. Лягушка-путешественница

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1.5 секунд
Ограничение по памяти:	512 мегабайт

Лягушонок Горф отправился в путешествие по Болотному королевству. К несчастью, он не рассчитал длину своего прыжка и упал в колодец глубиной n метров. Теперь Горф лежит на самом дне колодца и ему предстоит долгий путь вверх.

Стенки колодца в некоторых местах скользкие, а в некоторых, наоборот, очень удобные. А именно, если Горф сейчас находится на глубине x метров от уровня земли, то за один прыжок он может подняться на любое расстояние от 0 до a_x метров вверх.

К сожалению, Горф не может прыгать без перерывов. После каждого прыжка ему нужно отдохнуть. Если он отдохнет на глубине x метров от уровня земли, то за это время проскользит вниз на b_x метров.

Определите, за какое минимальное число прыжков Горф способен подняться до уровня земли.

Формат входных данных

В первой строке задано целое положительное число n ($1 \leq n \leq 300\,000$) — глубина колодца.

Во второй строке задано n целых чисел $a_1, a_2, \dots, a_n$ ($0 \leq a_i \leq i$) — максимальная высота, доступная для прыжка с каждой глубины.

В третьей строке вводится n целых чисел $b_1, b_2, \dots, b_n$ ($0 \leq b_i \leq n - i$) — глубина, на которую лягушонок провалится, взяв перерыв на каждой из глубин.

Формат выходных данных

В первой строке выведите целое число k — минимально возможное количество прыжков. В случае, если лягушонок не может выбраться из колодца, выведите -1 .

Если лягушонок мог выбраться из колодца, то во второй строке выведите последовательность глубин длины k , на которые он будет запрыгивать каждым своим прыжком. Считайте глубину вне колодца в любой точке равной нулю.

Если существует несколько решений, выведите любое из них.

Примеры

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

Замечание

В первом тесте из условия Горф находится на дне колодца, за один прыжок вверх поднимается к отметке в 1 метр глубины. После этого он соскальзывает вниз на метр и оказывается на отметке в 2 метра. С этой отметки он уже сможет выпрыгнуть из колодца за один прыжок.

Во втором тесте из условия Горф может допрыгнуть до отметки в один метр, но сразу после этого соскользнет обратно на дно колодца, поэтому ему не выбраться.

В третьем тесте из условия выбраться из колодца можно только прыгнув с глубины 5. Попасть напрямую туда нельзя, но можно добраться с помощью серии прыжков $10 \rightarrow 9 \rightarrow 4 \dashrightarrow 5$, где прыжок вверх обозначается простой стрелочкой, а пунктиром обозначен спуск

Задача К. Транспортировка

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

К очередной Летней компьютерной школе было решено подготовить кружки как для школьников, так и для всех преподавателей.

Имея привычку делать важные дела в самый последний момент, дизайнер закончил работу над макетом за два дня до начала школы. Ещё день уйдёт у завода-изготовителя на то, чтобы изготовить кружки и нанести на них изображение. На то, чтобы довезти кружки от завода-изготовителя до ЛКШ, остаётся всего 24 часа.

Заказ на 10000000 экземпляров кружек (а именно столько заказали организаторы), конечно же, за один рейс не увезти. Однако, за первый рейс хочется привезти максимальное количество кружек. Для перевозки был заказан один большегрузный автомобиль. Но есть один нюанс: на некоторых дорогах установлено ограничение на вес автомобиля. Поэтому если автомобиль нагрузить кружками под завязку, то, возможно, не удастся воспользоваться самым коротким маршрутом, а придётся ехать в объезд. Может случиться даже так, что из-за этого грузовик не успеет доехать до лагеря вовремя, а этого допустить никак нельзя. Итак, сколько же кружек можно погрузить в автомобиль, чтобы успеть привезти этот ценный груз вовремя, и не нарушая правил дорожного движения?

Формат входных данных

В первой строке находятся числа n ($1 \leq n \leq 500$) и m - количество узловых пунктов дорожной схемы и количество дорог, соответственно. В следующих m строках находится информация о дорогах. Каждая дорога описывается в отдельной строке следующим образом. Сначала указаны номера узловых пунктов, которые соединяются данной дорогой, потом время, которое тратится на проезд по этой дороге, и, наконец, максимальный вес автомобиля, которому разрешено ехать по этой дороге. Известно, что все дороги соединяют различные пункты, причем для каждой пары пунктов есть не более одной дороги, непосредственно их соединяющей. Все числа разделены одним или несколькими пробелами.

Узловые пункты нумеруются числами от 1 до n . При этом завод по производству кружек имеет номер 1, а ЛКШ - номер n . Время проезда по дороге задано в минутах и не превосходит 1440 (24 часа). Ограничение на массу задано в граммах и не превосходит одного миллиарда. Кроме того, известно, что одна кружка весит 100 грамм, а пустой грузовик - 3 тонны.

Формат выходных данных

Выведите одно число - максимальное количество кружек, которое можно привезти за первый рейс, 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 L. 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

Задача М. 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 дойти 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 дойти 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.