

Задача А. Задача для второклассника.

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

У этой задачи будет математическое условие, так как мне лень придумывать легенду.

Дано семейство графов G_1, \dots, G_k . Все они обладают общим множеством вершин $V = \{1, \dots, n\}$. Известно, что в этих графах могут встречаться m различных ребер, которые описываются следующим образом.

- i -е ребро встречается в графах $G_{l_1}, \dots, G_{l_{r_i}}$ и соединяет в каждом из них вершины a_i и b_i .

Для каждого графа G_i вычислите количество компонент связности, из которых он состоит.

Формат входных данных

В первой строке указаны числа n , m и k ($1 \leq n, m, k \leq 300\,000$) — количество вершин, ребер и графов.

В каждой из последующих m строк содержится описание соответствующего ребра, которое состоит из чисел a, b, l, r ($1 \leq a, b \leq n, 1 \leq l \leq r \leq k, a \neq b$).

Формат выходных данных

В единственной строке выведите k чисел, где i -е число равно количеству компонент связности в графе G_i .

Примеры

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

Задача В. Дороги

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

Петя недавно стал депутатом и теперь строит дороге в городе E , город состоит из n перекрестков, которые могут быть соединены дорогами. Есть m предложений от транспортных компаний, каждое из которых устроено следующим образом:

- Компания строит дорогу между городами a и b ;
- Предложение можно принять с l по r день;
- После постройки этой дороги радость горожан увеличится на w .

Так как дороги будут строиться за государственный счет, то не должно быть построено лишних дорог: дорога считается лишней, если ее удаление не влияет на связность города.

Петя может принять решение о постройке дорог в какой из дней с 1-го по k -й и заключить договора с компаниями, которые предоставляют свое предложение в этот день (возможно, не со всеми). Для каждого из этих дней выведите наибольший показатель счастья горожан, которого может добиться Петя, если заключит все договоры в этот день.

Формат входных данных

В первой строке указаны числа n , m и k ($1 \leq n, m, k \leq 300\,000$) — количество перекрестков, количество предложений и количество дней, в которые можно заключить договор.

Каждая из последующих m строк содержит по пятерке чисел a_i, b_i, l_i, r_i, w_i ($1 \leq a_i, b_i \leq n$, $a \neq b$, $1 \leq l \leq r \leq k$, $1 \leq w_i \leq 10^9$) — номера перекрестков, которые соединяет i -я дорога, первый и последний день, когда доступно i -е предложение и количество, на которое увеличится счастье горожан, если этот договор будет заключен.

Формат выходных данных

В единственной строке выведите k чисел — ответ на задачу.

Пример

стандартный ввод	стандартный вывод
3 3 5 1 2 1 4 2 2 3 2 4 2 1 3 3 3 8	2 4 10 4 0

Замечание

Если вы считаете, что в этой задаче плохие семплы, то пишите свои предложения в телеграмм `teraqqq`. Предложения без конкретных тестов будут отклоняться.

Задача С. Динамический Co-SAT

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

Little Саша играл с карточками в детском садике. Он заметил, что у каждой карточки есть две стороны, и для каждой пары карточек известно, что они образуют красивую пару, если хотя бы одна карточка лежит кверху нужной стороной... В общем, Саша поиграл-поиграл и, спустя пару очевидных переходов, придумал задачку про графы.

Дан граф на n вершинах и m ориентированных ребрах. Для каждого x от 1 до m вычислите количество компонент сильной связности в графе, если из него удалить все ребра с номерами больше, чем x .

Формат входных данных

В первой строке указана пара чисел n и m ($1 \leq n, m \leq 300\,000$) — количество вершин и ребер в графе, соответственно.

В последующих m строках указаны пары чисел a_i, b_i ($1 \leq a_i, b_i \leq n, a_i \neq b_i$) — начало и конец i -го ребра.

Формат выходных данных

Выведите n чисел — ответ на задачу.

Примеры

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

Задача D. И тут вы непредставляете, товарищ следователь...

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

На одной из детских образовательных смен школьники решили сыграть в игру на графе. Играть на бумажке или компьютере им показалось неинтересным занятием, поэтому они решили сыграть на дорогах города Ельца. В ходе игры один из школьников разрушил одну из дорог и нарушил связность города. Теперь всех школьников задержали, и с ними разговаривает следователь.

Всего на смене было n школьников, и изначально каждый будет говорить следователю, что это он во всем виноват. Следователь будет допрашивать школьников t часов. Каждый час описывается тройкой чисел (p, x, y) , где $1 \leq p, x, y \leq n$ и происходит следующим образом:

- Следователь говорит со школьником $p = p_1$ и выслушивает его мнение, кто виноват. Если школьник сказал, что виноват он сам, то следователь останавливает допрос. В ином случае, если школьник p_1 сказал, что виноват школьник p_2 , то следователь уже говорит со школьником p_2 и так далее. Процесс останавливается либо, когда следователь нашел сознавшегося школьника, либо когда он опрашивает какого-то школьника второй раз. В первом случае следователь записывает номер сознавшегося школьника и количество школьников, которых он опросил. Во втором случае следователь ставит просто прочерк;
- После столь утомительного процесса школьнику с номером x становится скучно, и в последующих часах он будет говорить следователю, что виноват школьник y (возможно, до того момента, как он снова не понимает свое мнение).

Вы преподаватель этой смены, у которого в этот день выходной. Выходной — событие радостное, но делать вам нечего, и развлечения ради вы решили посоревноваться со следователем в скорости. Поэтому ваша цель определить, какого школьника следователь запишет в бланк для каждого из t часов допроса.



Формат входных данных

В первой строке указаны числа n и t ($1 \leq n, t \leq 3 \cdot 10^5$) — количество школьников и количество часов допроса, соответственно.

Каждая из m последующих строк содержит по три целых числа p, x и y ($1 \leq p, x, y \leq n$), которые описывают очередной час допроса.

Формат выходных данных

В m строках выведите результат для каждого часа допроса. Если следователь нашел сознавшегося школьника, то выведите номер этого школьника, а также количество опрошенных школьников. Если следователь поставил прочерк, то выведите -1 .

Пример

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

Задача Е. Почти Link Cut

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

Дан набор подвешенных деревьев. Требуется выполнять следующие операции:

- 0 u v Для двух заданных вершин u и v выяснить, лежат ли они в одном дереве. Если это так, вывести вершину, являющуюся их наименьшим общим предком, иначе вывести 0.
- 1 u v Для корня u одного из деревьев и произвольной вершины v другого дерева добавить ребро (v, u) . В результате эти два дерева соединятся в одно.

Вам необходимо выполнять все операции online, т.е. вы сможете узнать следующий запрос только выполнив предыдущий.

Формат входных данных

На первой строке входного файла находится число n — суммарное количество вершин в рассматриваемых деревьях, $1 \leq n \leq 50000$. На следующей строке расположено n чисел — предок каждой вершины в начальной конфигурации, или 0, если соответствующая вершина является корнем. Затем следует число k — количество запросов к вашей программе, $1 \leq k \leq 100000$. Каждая из следующих строк содержит по три целых числа: вид запроса (0 — для поиска LCA или 1 — для добавления ребра) и два числа x, y . Вершины, участвующие в запросе можно вычислить по формуле: $u = (x - 1 + ans) \bmod n + 1$, $v = (y - 1 + ans) \bmod n + 1$, где ans - ответ на последний запрос типа 0 ($ans = 0$ для первого запроса).

Формат выходных данных

Для каждого запроса типа 0, выведите в выходной файл одно число на отдельной строке — ответ за этот запрос.

Пример

стандартный ввод	стандартный вывод
5	0
0 0 0 0 0	5
12	5
1 5 3	3
0 2 5	2
1 4 2	3
1 1 5	3
0 1 5	2
1 3 4	
0 1 5	
0 3 1	
0 4 2	
0 1 4	
0 5 2	
0 4 1	

Задача F. Парк

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

В столице Байтленда есть огороженный парк, территория которого представляет собой прямоугольник. Деревья и посетители в парке представлены в виде кругов.

В парке четыре входа, по одному в каждом углу (1 — внизу слева, 2 — внизу справа, 3 — вверху справа, 4 — вверху слева). Посетители могут входить и выходить из парка только через входы. Посетители могут входить и выходить из парка, когда они касаются обеих сторон угла соответствующего входа. Посетители могут свободно перемещаться по парку, но они не могут перекрывать деревья и заборы.

Ваша задача — рассчитать для каждого посетителя, учитывая вход, в который они войдут в парк, через какие выходы они могут выйти из парка.

Формат входных данных

Первая строка ввода содержит два целых числа n и m ($1 \leq n \leq 2000$ $1 \leq m \leq 100\,000$) — количество деревьев в парке и количество посетителей.

Вторая строка ввода содержит два целых числа w и h ($1 \leq w, h \leq 10^9$) — ширину и высоту парковой зоны. Нижний левый угол имеет координаты $(0, 0)$, а верхний правый угол имеет координаты (w, h) .

После этого есть n строк, которые описывают деревья. Каждая строка содержит три целых числа x , y и r , что означает, что центр i -го дерева (x, y) и его радиус r . Деревья не перекрывают друг друга или забор.

Наконец, есть m строк, которые описывают посетителей. Каждая строка содержит два целых числа r и e — радиус посетителя и вход, в который они войдут в парк.

Кроме того, ни одно дерево не перекрывает квадратную область $2k \times 2k$ в каждом углу, где k — радиус наибольшего посетителя.

Формат выходных данных

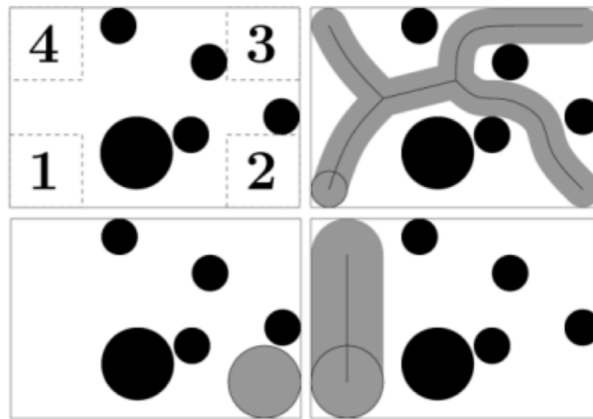
Вы должны вывести для каждого посетителя одну строку, содержащую входы, через которые они могут выйти из парка, в отсортированном порядке без пробелов между ними.

Пример

стандартный ввод	стандартный вывод
5 3	1234
16 11	2
11 8 1	14
6 10 1	
7 3 2	
10 4 1	
15 5 1	
1 1	
2 2	
2 1	

Замечание

На следующем рисунке показаны зоны входа и возможные маршруты для каждого посетителя:



Задача G. Раскрась одновременно

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 6 секунд
Ограничение по памяти: 512 мегабайт

Задана матрица, состоящая из n строк и m столбцов.

Вы можете выполнять над ней два типа действий:

- покрасить весь столбец в синий;
- покрасить всю строку в красный.

Обратите внимание, что вы не можете выбирать, в какой цвет красить строку или столбец.

За одну секунду можно выполнить как одно действие, так и несколько одновременно. Если сделать одно действие, то это будет бесплатно. Если же сделать $k > 1$ действий одновременно, то на это придется потратить k^2 монет. Когда несколько действий выполняются одновременно, для каждой клетки, которая затрагивается действиями обоих типов, цвет можно выбрать независимо.

Требуется обработать q запросов. Перед каждым запросом все ячейки становятся бесцветными. Изначально ограничений на итоговый цвет клеток нет. В i -м запросе добавляется ограничение вида:

- $x_i y_i c_i$ — клетка в строке x_i в столбце y_i должна быть покрашена в цвет c_i .

Таким образом, после i запросов существует i ограничений на необходимые цвета клеток матрицы. После каждого запроса выведите минимальную стоимость покраски в соответствии с ограничениями.

Формат входных данных

В первой строке записано три целых числа n, m и q ($1 \leq n, m, q \leq 2 \cdot 10^5$) — размер матрицы и количество запросов.

В i -й из следующих q строк записаны два целых числа x_i, y_i и символ c_i ($1 \leq x_i \leq n; 1 \leq y_i \leq m; c_i \in \{‘R’, ‘B’\}$, где ‘R’ значит красный, а ‘B’ значит синий) — описание i -го ограничения. Клетки во всех ограничениях попарно различные.

Формат выходных данных

Выведите q целых чисел — После каждого запроса выведите минимальную стоимость покраски в соответствии с ограничениями.

Примеры

стандартный ввод	стандартный вывод
2 2 4	0
1 1 R	0
2 2 R	0
1 2 B	16
2 1 B	
3 5 10	0
1 1 B	0
2 5 B	0
2 2 B	0
2 3 R	0
2 1 B	0
3 2 R	16
3 3 B	16
1 2 R	25
1 3 B	25
3 1 B	

Задача N. Депрессивная дружба

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

В классе есть n детей, известно, что среди них есть $n - 1$ пара друзей. При том дружеские связи образуют собой структуру дерева.

Но в этом классе не все так радужно, как может показаться на первый взгляд. На i день i ребенок уходит в депрессию и больше никогда не возвращается в класс. Тогда все его друзья собираются, обсуждают, что же такое произошло, и все они в честь этого становятся попарными друзьями.

Также в этой школе считают для каждого класса следующую характеристику: количество упорядоченных троек различных детей (a, b, c) удовлетворяющим следующим условиям: Никто из них не в депрессии, дети a и b дружат, а также дети b и c дружат.

И классному руководителю нужно собирать эту статистику каждый день, но так как у него и так много обязанностей, да и детей не мало, то он просит вас помочь ему, и для каждого i сказать, сколько было подходящих троек непосредственно до ухода i ребенка в депрессию.

Формат входных данных

В первой строке задано число n ($2 \leq n \leq 2 \cdot 10^5$) — количество детей.

Затем в следующих $n - 1$ строке записаны по дружеской паре a, b ($1 \leq a, b \leq n$), $a \neq b$.

Формат выходных данных

Выведите n строк, где на i строке количество подходящих троек, до ухода i ребенка в депрессию

Примеры

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

Задача I. Шарады

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1.5 секунд
Ограничение по памяти:	64 мегабайта

В ещё не изведанной части вселенной есть планета, на которой живут одни математики. На этой планете живут N математиков, каждый — в своём городе. Никакие два города не соединены дорогами, потому что математики могут общаться онлайн, оставляя комментарии о научных трудах друг друга.

Всё шло тихо и спокойно, пока один математик не решил написать научную работу со своего мобильного телефона. Автоисправление в телефоне заменило «очевидно» на «шарада». Не перечитав свою работу, математик так и опубликовал её. Совсем скоро об игре в шарады узнали все математики планеты, и им захотелось собраться и поиграть всем вместе. Поэтому в скором времени началась постройка дорог между городами. Строительство дорог будет идти M дней в соответствии со следующим расписанием: в первый день строятся дороги между всеми парами городов, у номеров которых наибольший общий делитель равен M . Во второй день строятся дороги между всеми парами городов, наибольший делитель номеров которых равен $M - 1$. И так далее до M -го дня, в который дороги строятся между всеми парами городов с взаимно простыми номерами. Говоря более формально, в i -й день (нумеруя дни с единицы) дороги строятся между всеми такими парами городов A и B , что $\text{НОД}(A, B) = M + 1 - i$.

Математики очень заняты постройкой дорог, поэтому они просят вас помочь определить минимальное число дней с начала строительства, через которое данная пара математиков сможет встретиться, чтобы поиграть в шарады.

Формат входных данных

В первой строке даны три целых положительных числа N , M и Q ($1 \leq N, Q \leq 100\,000$, $1 \leq M \leq N$) — количество городов, длительность строительства дорог и количество запросов соответственно.

В следующих Q строках вводятся по два целых числа A и B ($1 \leq A, B \leq N$) — номера городов двух математиков, которым интересно, через сколько дней они смогут встретиться (добраться из одного города в другой, проехав по уже построенным дорогам).

Формат выходных данных

На каждый из Q запросов выведите ответы — Q чисел, каждое в отдельной строке.

Примеры

стандартный ввод	стандартный вывод
8 3 3 2 5 3 6 4 8	3 1 2
25 6 1 20 9	4
9999 2222 2 1025 2405 3154 8949	1980 2160

Замечание

Пояснение к первому тесту:

В первый день строится дорога $(3, 6)$. Поэтому ответ на второй запрос 1. На второй день строятся дороги $(2, 4)$, $(2, 6)$, $(2, 8)$, $(4, 6)$ и $(6, 8)$. Города 4 и 8 теперь связаны (можно добраться из первого во второй используя город 6). На третий день строятся дороги между взаимно простыми городами, поэтому города 2 и 5 оказываются соединены.

Пояснение ко второму тесту:

На второй день строится дорога $(20, 15)$, на четвертый день — дорога $(15, 9)$. Таким образом, начиная с четвертого дня, города, 20 и 9 связаны (через город 15).

Задача J. Блогеры-путешественники

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	3 секунды
Ограничение по памяти:	1024 мегабайта

Ян и Татьяна решили стать блогерами-путешественниками и публиковать ролики о поездках по городам своей страны.

В стране есть n городов, пронумерованных от 1 до n . Город 1 — столица их страны. Города соединены m двусторонними дорогами, пронумерованными от 1 до m , каждая из которых соединяет два различных города. При этом одну и ту же пару городов могут соединять несколько различных дорог. Из любого города по дорогам можно доехать до любого другого города страны.

Путешественники планируют отправиться из столицы в какой-то другой город, но пока не выбрали в какой. Маршрут путешествия в город k будет состоять из городов s_1, s_2, \dots, s_q и дорог r_1, r_2, \dots, r_{q-1} , таких что:

- $s_1 = 1, s_q = k$;
- дорога r_i соединяет города s_i и s_{i+1} ;
- ребята не проезжают по одной и той же дороге дважды, поэтому все r_i различны. Допускается проезжать несколько раз через один и тот же город, в том числе через город 1, где путешествие начинается, и город k , в котором путешествие заканчивается.

Для каждой дороги Ян и Татьяна посчитали длительность ролика, который получится при съемке путешествия по этой дороге, длительность ролика для дороги с номером i равна t_i .

В процессе путешествия каждый из ребят выберет одну из дорог маршрута и снимет ролик, посвящённый этой дороге. При этом Ян любит снимать короткие ролики, поэтому выберет на маршруте дорогу с наименьшим значением t_i , а Татьяна предпочитает длинные ролики, поэтому выберет дорогу с наибольшим значением t_i .

Суммарная длина двух роликов будет равна $\min_{1 \leq i \leq q-1} t_{r_i} + \max_{1 \leq i \leq q-1} t_{r_i}$.

Ребята планируют выложить ролики на известную платформу, где большей популярностью пользуются короткие ролики, поэтому они хотят минимизировать суммарную длину двух роликов. Чтобы выбрать конечный город и маршрут для путешествия, блогеры хотят для каждого конечного города k подсчитать минимальную по всем возможным маршрутам из города 1 в город k суммарную длину двух роликов.

Формат входных данных

В первой строке даны два целых числа n, m ($2 \leq n \leq 300\,000, 1 \leq m \leq 300\,000$) — количество городов и дорог.

Следующие m строк содержат описания дорог. В i -й из этих строк находятся три целых числа u_i, v_i, t_i ($1 \leq u_i, v_i \leq n, u_i \neq v_i, 0 \leq t_i \leq 10^9$) — номера городов, соединённых дорогой, и длительность ролика про эту дорогу.

Гарантируется, что по имеющимся дорогам можно проехать из любого города в любой другой, возможно, через другие города.

Формат выходных данных

Для каждого $2 \leq k \leq n$ выведите минимальную суммарную длину роликов Яна и Татьяны для путешествия, заканчивающегося в городе k .

Примеры

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

Замечание

В первом примере возможные оптимальные маршруты:

- $1 \xrightarrow{t=1} 3 \xrightarrow{t=1} 2$. Длина роликов в маршруте $1 + 1 = 2$.
- $1 \xrightarrow{t=1} 3$. Длина роликов в маршруте $1 + 1 = 2$.

Во втором примере возможные оптимальные маршруты:

- $1 \xrightarrow{t=2} 2$. Длина роликов в маршруте $2 + 2 = 4$.
- $1 \xrightarrow{t=2} 2 \xrightarrow{t=3} 3$. Длина роликов в маршруте $2 + 3 = 5$.
- $1 \xrightarrow{t=2} 2 \xrightarrow{t=3} 3 \xrightarrow{t=4} 5 \xrightarrow{t=4} 4$. Длина роликов в маршруте $2 + 4 = 6$.
- $1 \xrightarrow{t=2} 2 \xrightarrow{t=3} 3 \xrightarrow{t=4} 5$. Длина роликов в маршруте $2 + 4 = 6$.
- $1 \xrightarrow{t=2} 2 \xrightarrow{t=3} 3 \xrightarrow{t=4} 5 \xrightarrow{t=4} 4 \xrightarrow{t=4} 6$. Длина роликов в маршруте $2 + 4 = 6$.
- $1 \xrightarrow{t=2} 2 \xrightarrow{t=8} 1 \xrightarrow{t=6} 7$. Длина роликов в маршруте $2 + 8 = 10$.

В третьем примере возможные оптимальные маршруты:

- $1 \xrightarrow{t=2} 2 \xrightarrow{t=0} 3 \xrightarrow{t=1} 4 \xrightarrow{t=3} 2$. Длина роликов в маршруте $0 + 3 = 3$.
- $1 \xrightarrow{t=2} 2 \xrightarrow{t=0} 3$. Длина роликов в маршруте $0 + 2 = 2$.
- $1 \xrightarrow{t=2} 2 \xrightarrow{t=0} 3 \xrightarrow{t=1} 4$. Длина роликов в маршруте $0 + 2 = 2$.