

Задача А. Остовное дерево

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

Найти в связном графе остовное дерево минимального веса.

Формат входных данных

Первая строка входного файла содержит два натуральных числа n и m — количество вершин и ребер графа соответственно ($1 \leq n \leq 20000, 0 \leq m \leq 100000$). Следующие m строк содержат описание ребер по одному на строке. Ребро номер i описывается тремя натуральными числами b_i , e_i и w_i — номера концов ребра и его вес соответственно ($1 \leq b_i, e_i \leq n, 0 \leq w_i \leq 100000$).

Граф является связным.

Формат выходных данных

Выведите единственное целое число — вес минимального остовного дерева.

Пример

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

Задача В. Реструктуризация компании

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

В жизни даже самой успешной компании может наступить кризисный период, когда приходится принимать тяжёлое решение о реструктуризации, распускать и объединять отделы, увольнять работников и заниматься прочими неприятными делами. Рассмотрим следующую модель компании.

В Большой Софтверной Компании работают n человек. Каждый человек принадлежит какому-то *отделу*. Исходно каждый человек работает над своим проектом в своём собственном отделе (таким образом, в начале компания состоит из n отделов по одному человеку).

Однако, в жизни компании наступили тяжёлые времена, и руководство было вынуждено нанять кризисного менеджера, который начал переустраивать рабочий процесс для повышения эффективности производства. Обозначим за $team(person)$ команду, в которой работает человек $person$. Кризисный менеджер может принимать решения двух типов:

1. Объединить отделы $team(x)$ и $team(y)$, сформировав из них один большой отдел, содержащий всех сотрудников $team(x)$ и $team(y)$, где x и y ($1 \leq x, y \leq n$) — номера каких-то двух сотрудников компании. Если $team(x)$ совпадает с $team(y)$, ничего делать не требуется.
2. Объединить отделы $team(x), team(x + 1), \dots, team(y)$, где x и y ($1 \leq x \leq y \leq n$) — номера каких-то двух сотрудников компании.

При этом кризисный менеджер иногда может интересоваться, работают ли в одном отделе сотрудники x и y ($1 \leq x, y \leq n$).

Помогите кризисному менеджеру, ответив на все его запросы.

Формат входных данных

Первая строка входных данных содержит два целых числа n и q ($1 \leq n \leq 200\,000, 1 \leq q \leq 500\,000$) — количество сотрудников компании и количество запросов кризисного менеджера.

В последующих q строках находятся запросы кризисного менеджера. Каждый запрос имеет вид $type\ x\ y$, где $type \in \{1, 2, 3\}$. Если $type = 1$ или $type = 2$, то запрос представляет собой решение кризисного менеджера об объединении отделов соответственно первого или второго вида. Если $type = 3$, то требуется определить, работают ли в одном отделе сотрудники x и y . Обратите внимание, что x может равняться y в запросе любого типа.

Формат выходных данных

На каждый запрос типа 3 выведите «YES» или «NO» (без кавычек), в зависимости от того, работают ли в одном отделе соответствующие люди.

Примеры

стандартный ввод	стандартный вывод
8 6	NO
3 2 5	YES
1 2 5	YES
3 2 5	
2 4 7	
2 1 2	
3 1 7	

Задача С. Подсчет опыта

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта

В очередной онлайн игре игроки, как обычно, сражаются с монстрами и набирают опыт. Для того, чтобы сражаться с монстрами, они объединяются в кланы. После победы над монстром, всем участникам клана, победившего его, добавляется одинаковое число единиц опыта. Особенностью этой игры является то, что кланы никогда не распадаются и из клана нельзя выйти. Единственная доступная операция — объединение двух кланов в один.

Поскольку игроков стало уже много, вам поручили написать систему учета текущего опыта игроков.

Формат входных данных

В первой строке входного файла содержатся числа n ($1 \leq n \leq 200000$) и m ($1 \leq m \leq 200000$) — число зарегистрированных игроков и число запросов.

В следующих m строках содержатся описания запросов. Запросы бывают трех типов:

- `join X Y` — объединить кланы, в которые входят игроки X и Y (если они уже в одном клане, то ничего не меняется).
- `add X V` — добавить V единиц опыта всем участникам клана, в который входит игрок X ($1 \leq V \leq 100$).
- `get X` — вывести текущий опыт игрока X .

Изначально у всех игроков 0 опыта и каждый из них состоит в клане, состоящим из него одного.

Формат выходных данных

Для каждого запроса `get X` выведите текущий опыт игрока X .

Пример

стандартный ввод	стандартный вывод
3 6	150
add 1 100	0
join 1 3	50
add 1 50	
get 1	
get 2	
get 3	

Задача D. День Объединения

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

В Бейтландии есть целых n городов, но нет ни одной дороги. Король страны, Вальдемар де Беар, решил исправить эту ситуацию и соединить некоторые города дорогами так, чтобы по этим дорогам можно было добраться от любого города до любого другого. Когда строительство будет завершено, король планирует отпраздновать День Объединения. К сожалению, казна Бейтландии почти пуста, поэтому король требует сэкономить деньги, минимизировав суммарную длину всех построенных дорог.

Формат входных данных

Первая строка входного файла содержит натуральное число n ($1 \leq n \leq 5000$) — количество городов в Бейтландии. Каждая из следующих n строк содержит по два целых числа x_i, y_i — координаты i -го города ($-10000 \leq x_i, y_i \leq 10000$). Никакие два города не расположены в одной точке.

Формат выходных данных

Первая строка выходного файла должна содержать минимальную суммарную длину дорог. Выведите число с точностью не менее 10^{-4} .

Пример

стандартный ввод	стандартный вывод
6	9.6568542495
1 1	
7 1	
2 2	
6 2	
1 3	
7 3	

Задача Е. Шарады

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

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 ($1 \leq N, Q \leq 100000, 1 \leq M \leq N$) — 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 ответ — Q чисел, каждое в отдельной строке.

Пример

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

Задача F. Постройка дорожной сети

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	128 мегабайт

Странствующий торговец прибыл в королевство Lazyland. Он планирует начать свое путешествие из столицы, посетить каждый из городов королевства, а затем вернуться в столицу. Торговец знает координаты каждого города, однако дороги еще не были построены. Очевидно, не имея дорог, не возможно посетить все города королевства!

Известно, что в королевстве есть n городов. Их координаты равны: $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$. Столица находится в точке (x_1, y_1) . Король выразил свое желание помочь торговцу. Он прикажет построить для него дороги. Однако, жители страны слишком ленивые.

Как и все жители страны, инженеры тоже очень ленивы, поэтому если потребуется построить дорогу от точки (x_i, y_i) до точки (x_j, y_j) , они будут использовать только горизонтальные или вертикальные отрезки. Поэтому нетрудно заметить, что длина такой дороги получится равной $|x_i - x_j| + |y_i - y_j|$.

Так как рабочие тоже невероятно ленивы, они согласятся построить не более, чем $n - 1$ дорогу для торговца — они слышали, что $n - 1$ дороги достаточно, чтобы путешествовать между всеми городами. Концами каждой дороги по понятным причинам будут какие-либо два различных города.

Торговцу дали право выбрать, какие именно дороги должны быть построены. Помогите ему вычислить длину минимального маршрута, который начинается в столице, обходит все города и затем возвращается в столицу при условии, что можно построить любые $n - 1$ дороги.

Формат входных данных

В первой строке записано число n ($1 \leq n \leq 10^4$) — количество городов в королевстве.

В каждой из следующих n строк записаны два числа x_i и y_i ($-1000 \leq x_i, y_i \leq 1000$) — координаты городов. Координаты столицы записаны в первой из этих строк.

Гарантируется, что никакие два города не находятся в одной точке.

Формат выходных данных

Выведите длину кратчайшего маршрута торговца, удовлетворяющего описанным в задаче требованиям.

Примеры

стандартный ввод	стандартный вывод
3 1 1 2 2 3 3	8
4 2 1 -1 2 -2 -1 1 -2	24
6 1 2 2 3 2 2 3 4 4 3 3 1	16

Задача G. Всем чмоки в этом чатике!

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

Сегодня Мэри, как программисту социальной сети «Телеграфчик», предстоит реализовать сложную систему управления чатами.

Задача Мэри усложняется тем, что в социальную сеть «Телеграфчик» внедрена продвинутая система шифрования «ZergRus», простая, как всё гениальное. Суть её в том, что в системе хранится одна переменная $zerg$, которая принимает значения от 0 (включительно) до $p = 10^6 + 3$ (исключая p) и меняется в зависимости от событий в системе.

В социальной сети всего n пользователей ($1 \leq n \leq 10^5$). В начале дня каждый пользователь оказывается в своём собственном чате, в котором больше никого нет. Переменная $zerg$ в начале дня устанавливается равной 0.

В течение дня происходят события типов:

1. Участник с номером $(i + zerg) \bmod n$ посылает сообщение всем участникам, сидящим с ним в чате (в том числе и себе самому), при этом переменная $zerg$ заменяется на $(30 \cdot zerg + 239) \bmod p$.
2. Происходит слияние чатов, в которых сидят участники с номерами $(i + zerg) \bmod n$ и $(j + zerg) \bmod n$. Если участники и так сидели в одном чате, то ничего не происходит. Если в разных, то чаты объединяются, а переменной $zerg$ присваивается значение $(13 \cdot zerg + 11) \bmod p$.
3. Участник с номером $(i + zerg) \bmod n$ хочет узнать, сколько сообщений он не прочитал, и прочитать их. Если участник прочитал q новых сообщений, то переменной $zerg$ присваивается значение $(100\,500 \cdot zerg + q) \bmod p$.

Вы поможете Мэри реализовать систему, обрабатывающую эти события?

Формат входных данных

В первой строке входного файла записаны натуральные числа n ($1 \leq n \leq 10^5$) — число пользователей социальной сети, и m ($1 \leq m \leq 3 \cdot 10^5$) — число событий, произошедших за день. В следующих m строках содержится описание событий. Первое целое число в строке означает тип события t ($1 \leq t \leq 3$). Если $t = 1$, далее следует число i ($0 \leq i < n$), по которому можно вычислить, какой участник послал сообщение. Если $t = 2$, далее следуют числа i и j ($0 \leq i, j < n$), по которым можно вычислить номера участников, чаты с которыми должны объединиться. Если $t = 3$, далее следует число i ($0 \leq i < n$), по которому можно вычислить номер участника, желающего узнать, сколько у него сообщений, и прочитать их.

Формат выходных данных

Для каждого события типа 3 нужно вывести число непрочитанных сообщений у участника.

Пример

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

Задача Н. Расстояние X

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 из N вершин и M ребер. Назовем ценой пути между двумя вершинами вес максимального ребра на этом пути. Найдите количество пар с минимальной ценой пути между ними, равной X .

Формат входных данных

Первая строка содержит три целых числа N , M и X ($1 \leq N \leq 10^5, 1 \leq M \leq 3 \cdot 10^5, 1 \leq X \leq 10^9$).
Следующие M строк содержат по три числа a_i , b_i и w_i , обозначающие ребро между вершинами a_i и b_i веса w_i . ($1 \leq w_i \leq 10^9$)

Формат выходных данных

Выведите одно число — ответ на задачу

Примеры

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

Задача I. Школы

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

С целью подготовки к проведению олимпиады по информатике мэр решил обеспечить надежным электроснабжением все школы города. Для этого необходимо провести линию электропередач от альтернативного источника электроэнергии "Майбутя" к одной из школ города (к какой неважно), а также соединить линиями электропередач некоторые школы между собой.

Считается, что школа имеет надежное электроснабжение, если она напрямую связана с источником "Майбутя" либо с одной из тех школ, которые имеют надежное электроснабжение.

Известна стоимость соединения между некоторыми парами школ. Мэр города решил выбрать одну из двух наиболее экономичных схем электроснабжения (стоимость схемы равняется сумме стоимостей соединений пар школ).

Напишите программу, которая вычисляет стоимость двух наиболее экономных схем альтернативного электроснабжения школ.

Формат входных данных

В первой строке входного файла находятся два натуральных числа, разделенных пробелом: N ($3 \leq N \leq 100$), количество школ в городе, и M — количество возможных соединений между ними.

В каждой из последующих M строк находятся по три числа: A_i, B_i, C_i , разделенных пробелами, где C_i — стоимость прокладки линии электроснабжения ($1 \leq C_i \leq 300$) от школы A_i до школы B_i .

Формат выходных данных

В единственной строке выходного файла должны содержаться два натуральных числа S_1 и S_2 , разделенных пробелом — две наименьшие стоимости схем ($S_1 \leq S_2$). $S_1 = S_2$ тогда и только тогда, когда существует несколько схем надежного электроснабжения наименьшей стоимости.

Гарантируется, что для входных данных существует две различные схемы надёжного электроснабжения.

Пример

стандартный ввод	стандартный вывод
5 8	110 121
1 3 75	
3 4 51	
2 4 19	
3 2 95	
2 5 42	
5 4 31	
1 2 9	
3 5 66	

Задача J. Еще одна задача про остов

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

Костя с Ваней при подготовке очередной задачи на персистентную двумерную динамическую структуру данных нашли массив a_i размера n , состоящий из целых чисел, каждое из которых находится в промежутке $[0, m - 1]$. Костя, как любитель цветных графов, сразу создал полный неориентированный граф из n вершин, вершине v был задан цвет a_v . Ваня, как любитель взвешенных графов, сразу задал каждому ребру (v, u) вес $f(a_v, a_u)$.

$$f(a, b) = \begin{cases} a + b & a + b < m \\ a + b - m, & \text{иначе} \end{cases}$$

Теперь ребят интересует вопрос: каким же будет вес минимального остовного дерева для нашего графа? Помогите им найти ответ.

Формат входных данных

Программа получает на вход два числа n, m ($1 \leq n \leq 2 \cdot 10^5, 1 \leq m \leq 10^9$).
В следующей строке вводится n целых чисел — массив a_i ($0 \leq a_i \leq m - 1$).

Формат выходных данных

Выведите одно число — суммарный вес остовного дерева.

Пример

стандартный ввод	стандартный вывод
3 2 0 1 1	1

Задача К. Плотное остовное дерево

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

Требуется найти в графе остовное дерево, в котором разница между весом максимального и минимального ребра минимальна.

Формат входных данных

Первая строка содержит два целых числа n и m ($n \leq 1\,000$, $m \leq 10\,000$) — количество вершин и ребер в графе.

Каждая из следующих m строк содержит три целых числа b_i , e_i и w_i ($1 \leq b_i, e_i \leq n$, $0 \leq |w_i| \leq 10^9$) — номера вершин, которые соединяет ребро, а также вес ребра.

Формат выходных данных

Если остовное дерево существует, в первой строке выведите YES, а во второй строке одно целое число — минимальную разность между весом максимального и минимального ребра в дереве.

В противном случае в единственной строке выведите NO.

Пример

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