

Вариант 3

1) #бинарный поиск, #структуры данных

В одном небезызвестном алгоритме нахождения k -порядковой статистики нужно разбить все элементы на пятерки подряд идущих элементов и найти медиану каждой пятерки. Медианой называют средний элемент отсортированного массива (для пятерки — это третий по величине элемент). Для оценки скорости работы этого алгоритма на современной видеокарте требуется уметь находить сумму медиан в каждой пятерке упорядоченного множества.

Суммой медиан отсортированного k -элементного множества $S = \{a_1, a_2, \dots, a_k\}$, где $a_1 < a_2 < a_3 < \dots < a_k$, будем называть

$$\sum_{\substack{i \leq k \\ i \bmod 5 = 3}} a_i.$$

Оператор **mod** обозначает взятие остатка, то есть $x \bmod y$ обозначает остаток при делении x на y .

Для проведения нагрузочного тестирования потребовалось быстро вычислять сумму медиан для изменяющегося множества.

Входные данные

В первой строке задано число n ($1 \leq n \leq 10^5$) — количество операций, производимых с множеством.

Далее в каждой из n строк описана одна из трех операций:

- `add x` — добавить в множество элемент x ;
- `del x` — удалить из множества элемент x ;
- `sum` — найти сумму медиан множества.

Для любой операции `add x` верно, что элемент x непосредственно перед операцией не входит в множество.

Для любой операции `del x` верно, что элемент x непосредственно перед операцией входит в множество.

Все числа во входных данных — положительные целые числа, не превосходящие 10^9 .

Выходные данные

Для каждой операции `sum` выведите на отдельной строке сумму медиан текущего множества. Если множество пусто, то выведите 0.

2) #система непересекающихся множеств, #графы, #поиск в глубину

У Леши много знакомых, и некоторые из них дружат друг с другом, а некоторые друг другу не нравятся. Чтобы вечеринка удалась на славу, Леша хочет пригласить только тех своих знакомых, которые дружат, и не приглашать тех, кто не нравится друг другу. Отношения дружбы и антипатии симметричны.

Более формально, для каждого приглашенного человека должны выполняться следующие условия:

- все его друзья должны быть также приглашены на вечеринку;
- среди приглашенных не должно быть людей, которые ему не нравятся;
- все приглашенные на вечеринку должны быть связаны с ним дружбой напрямую или через цепь общих друзей произвольной длины. Будем говорить, что люди a_1 и a_p связаны цепью общих друзей, если существует последовательность людей a_2, a_3, \dots, a_{p-1} такая, что все пары людей a_i и a_{i+1} ($1 \leq i < p$) — друзья.

Помогите Леше определить максимальное количество знакомых, которых он сможет пригласить.

Входные данные

В первой строке входных данных записано целое число n — количество знакомых Леши. ($1 < n < 200$)

Во второй строке записано целое число k ($0 \leq k \leq \min(10^5, \frac{n \cdot (n-1)}{2})$) — количество пар друзей. В следующих k строках через пробел записаны пары чисел u_i, v_i ($1 \leq u_i, v_i \leq n, u_i \neq v_i$) — номера людей, которые входят в i -ую пару друзей.

В следующей строке записано число m ($0 \leq m \leq \min(10^5, \frac{n \cdot (n-1)}{2})$) — количество пар людей, которые друг другу не нравятся. В следующих m строках перечислены пары таких людей в том же формате, что и пары друзей.

Каждая пара людей упоминается во входных данных не более одного раза ($0 \leq k + m \leq \frac{n \cdot (n-1)}{2}$). В частности, два человека не могут быть друзьями и одновременно не нравиться друг другу.

Выходные данные

Выведите единственное число — максимальное количество людей, которых Леша сможет пригласить на вечеринку.

3) #бинарный поиск

Саша планирует путешествие на космическом корабле, за время которого он посетит n планет. Для i -ой планеты известны числа a_i — максимальное число чемоданов, которые инопланетному туристу разрешено провезти на планету, и b_i — число жителей на планете.

В путешествие Саша возьмет с собой сувениры. Сувениры упаковываются в чемоданы, по x штук в каждый. На корабль Саша возьмет ровно $a_1 + \dots + a_n$ чемоданов.

Высаживаясь на i -ой планете, Саша берет с собой a_i чемоданов. В первый день пребывания на планете Саша гуляет и знакомится с жителями. Во второй и все последующие дни дарит жителям сувениры — каждому из b_i жителей по сувениру каждый день. Саша покидает планету вечером того дня, когда сувениров становится меньше, чем жителей (то есть как только на следующий день он не сможет подарить нужное количество сувениров). Оставшиеся сувениры он оставляет в отеле.

Всего Саша собирается путешествовать ровно c дней. Время, затраченное на перелеты между планетами, считается равным нулю. Сколькими способами можно выбрать целое положительное число x так, чтобы планируемое путешествие заняло ровно c дней?

Входные данные

Первая строка входных данных содержит целые числа n и c , разделенные пробелом, — число планет, которые Саша собирается посетить, и число дней, отведенных на путешествие, соответственно.

Следующие n строк содержат пары целых чисел a_i, b_i ($1 \leq i \leq n$), разделенных пробелом, — число чемоданов, разрешенных для ввоза на i -ю планету, и число жителей на i -ой планете, соответственно.

- $1 \leq n \leq 10^4$
- $0 \leq a_i \leq 10^9$
- $1 \leq b_i \leq 10^9$
- $1 \leq c \leq 10^9$

Выходные данные

Выведите единственное целое число k — количество способов выбрать x так, чтобы путешествовать ровно c дней. Если существует бесконечно много вариантов выбора x , выведите -1.

4) #дерево отрезков, #дерево Фенвика

Женя придумал новый вид шифрования сообщений и хочет проверить его работу. Делать это вручную долго и трудоемко, поэтому он решил обратиться к Вам.

Сообщение представляет собой n целых чисел a_1, a_2, \dots, a_n . Для шифрования используется ключ, который представляет собой m целых чисел b_1, b_2, \dots, b_m ($m \leq n$). Все числа из сообщения и из ключа лежат в интервале от 0 до $c - 1$, включительно, и все последующие вычисления проводятся по модулю c .

Шифрование проводится в $n - m + 1$ этапов. На первом этапе к каждому из чисел a_1, a_2, \dots, a_m прибавляются соответствующие числа b_1, b_2, \dots, b_m . На втором этапе к числам a_2, a_3, \dots, a_{m+1} (измененным на предыдущем этапе) прибавляются числа b_1, b_2, \dots, b_m . И так далее: на этапе номер i к числам $a_i, a_{i+1}, \dots, a_{i+m-1}$ прибавляются числа b_1, b_2, \dots, b_m . Результатом шифрования является последовательность a_1, a_2, \dots, a_n после $n - m + 1$ этапов шифрования.

Помогите Жене: напишите программу, которая будет осуществлять шифрование сообщений описанным способом.

Входные данные

Первая строка входных данных содержит три целых числа n, m и c , разделенных единичными пробелами.

Вторая строка входных данных содержит n целых чисел a_i ($0 \leq a_i < c$), разделенных единичными пробелами, — исходное сообщение.

Третья строка входных данных содержит m целых чисел b_i ($0 \leq b_i < c$), разделенных единичными пробелами, — ключ шифрования.

- $1 \leq m \leq n \leq 10^5$
- $1 \leq c \leq 10^3$

Выходные данные

Выведите n целых чисел, разделенных пробелами, — результат шифрования сообщения.

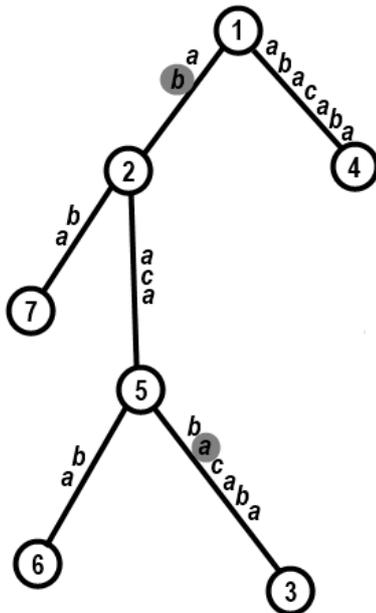
5) #деревья, #алгоритм Кнута-Морриса-Пратта, #поиск в глубину

В этой задаче корнем дерева будет вершина с номером 1.

Обозначим через $d(v, u)$ длину кратчайшего по количеству ребер пути в дереве между вершинами v и u .

Родителем вершины v в корневом дереве с корнем в вершине r ($v \neq r$) называется вершина p_v , такая, что $d(r, p_v) + 1 = d(r, v)$ и $d(p_v, v) = 1$. Например, на рисунке родителем вершины $v = 5$ является вершина $p_5 = 2$.

Как-то раз Поликарп раздобыл корневое дерево из n вершин. Дерево было не совсем обычное — на его ребрах были написаны строки. Поликарп расположил дерево на плоскости так, что все ребра ведут сверху вниз при прохождении от родителя вершины к вершине (см. рисунок). Для каждого ребра, ведущего из вершины p_v в вершину v ($1 < v \leq n$), известна строка s_v , которая на нем написана. Все строки записаны на ребрах сверху вниз. Например, на рисунке $s_7 = \text{«ba»}$. Символы в строках пронумерованы от 0.



Позицией в этом дереве Поликарп называет конкретную букву конкретной строки. Позиция записывается как пара целых чисел (v, x) , которая обозначает, что позицией является x -ая буква строки s_v ($1 < v \leq n$, $0 \leq x < |s_v|$), где $|s_v|$ — длина строки s_v . Например, выделенные буквы — это позиции $(2, 1)$ и $(3, 1)$.

Рассмотрим пару позиций (v, x) и (u, y) в дереве Поликарпа, такую, что путь от первой позиции ко второй идет сверху вниз в дереве Поликарпа, нарисованном на плоскости. Будем считать, что пара этих позиций определяет строку z . Строка z состоит из всех букв на пути от (v, x) к (u, y) , записанных в порядке прохождения этого пути. Например, на рисунке выделенные позиции определяют строку «bacaba».

У Поликарпа есть строка t , он хочет узнать, сколько существует пар позиций, которые определяют строку t . Обратите внимание, что путь от первой позиции ко второй в паре должен всюду вести вниз. Помогите ему с этой нелегкой древесно-строковой задачей!

Входные данные

В первой строке задано целое число n ($2 \leq n \leq 10^5$) — количество вершин дерева Поликарпа. В следующих $n - 1$ строках заданы ребра дерева. В i -ой из них записаны число p_{i+1} и строка s_{i+1} ($1 \leq p_{i+1} \leq n$; $p_{i+1} \neq (i + 1)$). Строка s_{i+1} — непустая и состоит из строчных букв латинского алфавита. В последней строке записана строка t . Строка t состоит из строчных букв латинского алфавита, ее длина не менее 2.

Гарантируется, что входные данные содержат не более $3 \cdot 10^5$ букв латинского алфавита.

Выходные данные

Выведите единственное целое число — искомое количество.

б) #деревья, #поиск в глубину, #поиск в ширину

Рекламное агентство "К" имеет чётко выраженную иерархическую структуру. В её вершине находится генеральный директор. Он имеет своих подчинённых, которые, в свою очередь, также могут иметь подчинённых и т.д. В один прекрасный день Таня придумала метод, помогающий увеличить отдачу от рекламы на 110%. Она сразу же позвонила своей начальнице, затем своей подруге Лене, находящейся у неё в подчинении, потом Кате и Маше. Те, в свою очередь, быстро перезвонили своим друзьям и т.д. Определите, какое наименьшее время понадобится для того, чтобы всё агентство "К" узнало о сказочно эффективном методе Тани. Но учтите, что телефоны рекламного агентства настроены так, чтобы каждый работник мог говорить либо со своим непосредственным начальником, либо со своим непосредственным подчинённым (иначе и нельзя — девушки прекрасно заменяют работу продолжительным общением по телефону, если им разрешить чуток больше). Каждый телефонный разговор занимает ровно одну минуту.

Входные данные

В первой строке находится одно число N ($N \leq 100000$) — количество служащих в фирме "К". Каждый служащий имеет свой уникальный ID — число от 1 до N .

Далее следует N строк. Строка с номером K содержит список подчинённых K -го служащего, завершающийся нулём. В последней строке находится одно число — Танин ID. Напомним, что в агентстве "К" у каждого сотрудника, кроме генерального директора, есть ровно один непосредственный начальник.

Выходные данные

Выведите одно число — минимальное время в минутах, за которое вся компания "К" узнает о гениальном методе простой девочки по имени Таня.

7) #деревья, #динамика на дереве

Компания А, занимающаяся грузоперевозками, имеет разветвлённую сеть филиалов. Бизнес идёт в гору, и компания решила организовать новые филиалы. Но прежде, чем это сделать, руководство компании решило проанализировать, насколько эффективно работает доставка товаров между филиалами. Для этого и требуются Ваш опыт и знания.

Схема доставки товаров между филиалами устроена так, что из любого филиала товар можно перевезти в любой другой (возможно через другие филиалы), причём единственным способом. Также известно $t[i, j]$ — время доставки товара между двумя филиалами (с номерами i и j), между которыми есть прямое сообщение. От Вас требуется определить среднее время на доставку товара между филиалами, то есть величину $\text{sum}(t[i, j]) / (N * (N - 1))$, где суммирование ведётся по $1 \leq i, j \leq N$ и $i \neq j$.

Входные данные

В первой строчке указано число N ($2 \leq N \leq 50000$) — количество филиалов в компании АСМ. Далее идёт $N-1$ строка, каждая из которых содержит тройку чисел a_i, b_i, c_i , где a_i, b_i ($1 \leq a_i, b_i \leq N$) — номера филиалов, имеющих прямое сообщение между собой, а c_i ($0 \leq c_i \leq 1000$; все c_i целые) — время доставки товара между этими филиалами.

Выходные данные

Выведите единственное число — среднее время доставки товара между филиалами с точностью 4 знака после десятичной точки.

8) #бинарный поиск, #строки

Вам задан поток логов о предупреждениях в программе. Каждая запись потока логов представляет собой строку в формате:

«2012-ММ-ДД HH:MM:SS:MESSAGE» (без кавычек).

Строка «MESSAGE» состоит из пробелов, заглавных и строчных букв латинского алфавита, а также символов «!», «.», «,», «?». Строка «2012-ММ-ДД» определяет корректную дату в 2012 году. Строка «HH:MM:SS» определяет корректное время в 24 часовом формате.

Описанная запись потока логов обозначает, что в определенное в записи время произошло некоторое предупреждение в программе (описание предупреждения содержит строка «MESSAGE»).

Ваша задача вывести первый момент времени, когда количество предупреждений за последние n секунд стало не менее m .

Входные данные

В первой строке входных данных записаны два целых числа через пробел n и m ($1 \leq n, m \leq 10000$).

Вторая и остальные строки входных данных представляют собой поток логов. Во второй строке входных данных записана первая запись потока логов, в третьей строке — вторая запись потока логов и так далее. Каждая запись потока логов имеет формат описанный выше. Все записи заданы в хронологическом порядке, то есть записи о предупреждениях заданы в порядке их появления в программе.

Гарантируется, что в логге есть хотя бы одна запись. Гарантируется, что суммарная длина всех строк потока логов не превышает $5 \cdot 10^6$ (в частности, это означает, что длина одной строки не превышает $5 \cdot 10^6$ символов). Гарантируется, что все заданные даты и времена корректны, а строка «MESSAGE» во всех записях не пустая.

Выходные данные

Если не существует искомого момента времени выведите -1. Иначе выведите строку в формате «2012-ММ-ДД HH:MM:SS» (без кавычек) — первый момент времени, когда количество предупреждений за последние n секунд стало не менее m .

9) #бинарный поиск

Даны три положительных целых числа x, y, n . Надо найти ближайшую дробь к дроби $\frac{x}{y}$, имеющую знаменатель, не превышающий n .

Формально, надо найти такую пару целых чисел a, b ($1 \leq b \leq n; 0 \leq a$), чтобы значение $\left| \frac{x}{y} - \frac{a}{b} \right|$ было минимально возможным.

Если есть несколько «ближайших» дробей, выберите дробь с наименьшим знаменателем.
Если есть несколько «ближайших» дробей с наименьшим знаменателем, выберите дробь с наименьшим числителем.

Входные данные

В единственной строке содержатся три целых числа x, y, n ($1 \leq x, y, n \leq 10^5$).

Выходные данные

Выведите необходимую дробь в формате « a/b » (без кавычек).

10) #бинарный поиск

На планете Т есть непонятные нам обычаи. Например, очень необычно для землян выбирают имена своим детям. Родители так выбирают имя ребенку, чтобы оно могло быть получено как удалением некоторого набора букв из имени отца, так и удалением некоторого набора букв из имени матери. Например, если отца зовут «abacaba», а мать — «bbccaa», то их ребенок может носить имена «a», «bba», «bcaa», но не может носить имена «aaa», «ab» или «bbc». Возможно, что имя ребенка совпадает с именем отца и/или матери, если оно может быть получено из имени другого родителя удалением нескольких (возможно, ни одной) букв.

Пусть отец по имени X и мать по имени Y выбирают имя своему новорожденному ребенку. Так как в школах учеников часто вызывают к доске в лексикографическом порядке имен учеников, то есть в порядке следования имен в словаре, то они хотят выбрать своему ребенку такое имя, чтобы оно лексикографически следовало как можно позже.

Формально, строка S лексикографически больше строки T, если выполняется одно из двух условий:

- строка T получается из S удалением одной или более букв с конца строки S;
- первые $(i - 1)$ символов строк T и S не различаются, а буква в i -й позиции строки T следует в алфавите раньше буквы в i -й позиции строки S.

Требуется написать программу, которая по именам отца и матери находит лексикографически наибольшее имя для их ребенка.

Входные данные

Первая строка содержит X — имя отца. Вторая строка содержит Y — имя матери. Каждое имя состоит из строчных букв латинского алфавита, включает хотя бы одну букву и имеет длину не более 105 букв.

Выходные данные

Результат должен содержать искомое лексикографически наибольшее из возможных имен ребенка. В случае, если подходящего имени для ребенка не существует, результат должен быть пустым