

Задача А. Директоры (50 баллов)

Ограничение по времени: 0.5 секунд
Ограничение по памяти: 64Mb

В маленькой корпорации изначально царила демократия: каждый работник был сам себе начальник, никто за его работой не следил и никто, кроме него самого, за нее не отвечал. Таких людей мы будем называть *директорами*.

Однако через некоторое время корпорация стала очень большой, и стало ясно, что так дальше продолжаться не может. Поэтому постепенно стали назначать одних сотрудников начальниками над другими. А именно, в некоторые моменты времени по очереди стали объявлять, что теперь сотрудник с идентификационным номером a является непосредственным начальником сотрудника с номером b . Причем если сотруднику уже однажды назначили непосредственного начальника, то этот начальник уже никогда не изменится.

Через некоторое время система отношений в корпорации стала сложной, и сотрудникам иногда становится интересно: кто же их чей директор. Иными словами рассмотрим сотрудника a , а затем его непосредственного начальника, а затем взять его непосредственного начальника и т.д. Гарантируется, что данный процесс когда-нибудь остановится на сотруднике b , который является директором, т.е. сам себе начальник. (Иначе говоря, в графе подчиненности нет циклов кроме петель.) Данного сотрудника b и будем называть директором для a .

Формат входных данных

В первой строке входа записаны числа n и m — количество сотрудников и количество запросов соответственно. Сотрудники пронумерованы числами от 1 до n . Запросы бывают двух типов: сделать сотрудника a непосредственным начальником сотрудника b или же спросить, кто является директором сотрудника a .

В каждой из следующих m строк — по одному запросу. Запрос первого типа задается двумя целыми числами $1 \leq a, b \leq n$. Запрос второго типа задается одним числом $1 \leq a \leq n$. При этом $1 \leq m, n \leq 100\,000$.

Формат выходных данных

Выведите ровно m чисел — по одному на запрос. Каждое число — в отдельной строке. Если запрос был сделать одного человека начальником другого, то нужно выдать 0 или 1: если запрос был корректным, т.е. подчиненному до этого не назначали еще начальников, и не образуется циклов в графе подчиненности (в том числе петель), то выдать 1, иначе выдать 0. В случае если запрос был корректен, необходимо выполнить назначение, иначе нужно его проигнорировать. Если запрос был узнать директора сотрудника с номером a , то нужно выдать номер того самого директора.

Замечание

Сложность алгоритма должна получиться $o(m \log n) + O(n)$. Обратите внимание, что первое o здесь малое, то есть алгоритм должен быть асимптотически лучше, чем $m \log n$.

Пример

| тест | ответ |
|------|-------|
| 4 22 | 1 |
| 1 | 2 |
| 2 | 3 |
| 3 | 4 |
| 4 | 1 |
| 1 2 | 0 |
| 3 2 | 1 |
| 1 | 1 |
| 2 | 1 |
| 2 3 | 1 |
| 1 | 1 |
| 2 | 1 |
| 3 | 1 |
| 3 4 | 1 |
| 1 | 1 |
| 2 | 1 |
| 3 | 1 |
| 4 | 0 |
| 4 1 | 1 |
| 1 | 1 |
| 2 | 1 |
| 3 | 1 |
| 4 | |

Задача В. Зашифрованное сообщение (75 баллов)

Ограничение по времени: 1 секунда
Ограничение по памяти: 64Mb

Вам дано зашифрованное сообщение Y , а также последовательность циклических сдвигов, с помощью которых зашифрованное сообщение было получено из исходного сообщения X . Каждый циклический сдвиг задан тремя параметрами i, j и k . Применить к текущей строке Z циклический сдвиг с параметрами (i, j, k) означает k раз циклически сдвинуть вправо подстроку $Z[i..j]$ (с i -го символа по j -й включительно). Т.е. если применить к строке $Z = z_1 z_2 z_3 z_4 z_5 z_6$ циклический сдвиг с параметрами $(2, 5, 3)$, то получится строка $z_1 z_3 z_4 z_5 z_2 z_6$. Символы в сообщениях нумеруются с единицы.

Имея такую информацию, вам нужно восстановить исходное сообщение.

Формат входных данных

В первой строке входа зашифрованное сообщение, которое является непустой строкой, состоящей из N маленьких латинских букв ($1 \leq N \leq 50\,000$). Вторая строка содержит количество сдвигов M ($1 \leq M \leq 50\,000$). Следующие M строк содержат последовательность циклических сдвигов (в порядке их применения к исходному сообщению). Каждый сдвиг описывается тремя i, j, k ($1 \leq i < j \leq N, 1 \leq k \leq j - i$).

Формат выходных данных

В единственную строку выхода выведите исходное сообщение.

Пример

| тест | ответ |
|------------------------------------------|----------|
| logoduck 3 1 3 1 4 5 1 1 4 1 | goodluck |

Задача С. Магия (80 баллов)

Ограничение по времени: 1 секунда
Ограничение по памяти: 64Mb

Сказочная страна представляет собой множество городов, соединенных дорогами с двухсторонним движением. Причем из любого города страны можно добраться в любой другой город либо непосредственно, либо через другие города. Известно, что в сказочной стране не существует дорог, соединяющих город сам с собой и между любыми двумя разными городами, существует не более одной дороги.

В сказочной стране живут желтый и синий волшебники. Желтый волшебник, пройдя по дороге, перекрашивает ее в желтый цвет, синий — в синий. Как известно, при наложении желтой краски на синюю, либо синей краски на желтую, краски смешиваются и превращаются в краску зеленого цвета, который является самым нелюбимым цветом обоих волшебников.

В этом году в столице страны (городе f) проводится конференция волшебников. Желтый и синий волшебники хотят узнать, какое минимальное количество дорог им придется перекрасить в зеленый цвет, чтобы добраться в столицу. Изначально все дороги не покрашены.

Начальное положение желтого и синего волшебников заранее не известно. Поэтому необходимо решить данную задачу для k возможных случаев их начальных расположений.

Формат входных данных

Первая строка входного файла содержит целые числа: n ($1 \leq n \leq 100\,000$) и m ($1 \leq m \leq 500\,000$) — количество городов и дорог в волшебной стране соответственно. Третья строка содержит одно целое число f ($1 \leq f \leq n$) — номер города, являющегося столицей сказочной страны. В следующих m строках, находится описание дорог страны. В этих m строк записано по два целых числа a_i и b_i , означающих, что существует дорога, соединяющая города a_i и b_i . Следующая строка содержит целое число k ($1 \leq k \leq 100\,000$) — количество возможных начальных расположений волшебников. Далее следуют k строк, каждая из которых содержит два целых числа — номера городов, в которых изначально находится желтый и синий волшебники соответственно.

Формат выходных данных

Для каждого из k случаев, ваша программа должна вывести в выходной минимальное количество дорог, которое придется покрасить в зеленый цвет волшебникам для того, чтобы добраться в столицу.

Пример

| тест | ответ |
|------|-------|
| 6 6 | 1 |
| 1 | 2 |
| 1 2 | |
| 2 3 | |
| 3 4 | |
| 4 2 | |
| 4 5 | |
| 3 6 | |
| 2 | |
| 5 6 | |
| 6 6 | |

Задача D. Опекуны карнотавров (75 баллов)

Ограничение по времени: 2 секунды
Ограничение по памяти: 64Mb

Карнотавры очень внимательно относятся к заботе о своем потомстве. У каждого динозавра обязательно есть старший динозавр, который его опекает. В случае, если опекуна съедают (к сожалению, в юрский период такое не было редкостью), забота о его подопечных ложится на плечи того, кто опекал съеденного динозавра. Карнотавры — смертоносные хищники, поэтому их обычаи строго запрещают им драться между собой. Если у них возникает какой-то конфликт, то, чтобы решить его, они обращаются к кому-то из старших, которому доверяют, а доверяют они только тем, кто является их опекуном или опекуном их опекуна и так далее (назовем таких динозавров суперопекунами). Поэтому для того, чтобы решить спор двух карнотавров, нужно найти такого динозавра, который является суперопекуном для них обоих. Разумеется, беспокоить старших по пустякам не стоит, поэтому спорщики стараются найти самого младшего из динозавров, который удовлетворяет этому условию. Если у динозавра возник конфликт с его суперопекуном, то этот суперопекун сам решит проблему. Если у динозавра нелады с самим собой, он должен разобраться с этим самостоятельно, не беспокоя старших. Помогите динозаврам разрешить их споры.

Формат входных данных

Во входном файле записано число M , обозначающее количество запросов ($1 \leq M \leq 200\,000$). Далее на отдельных строках следуют M запросов, обозначающих следующие события:

- $+ v$ — родился новый динозавр и опекунство над ним взял динозавр с номером v . Родившемуся динозавру нужно присвоить наименьший натуральный номер, который до этого еще никогда не встречался.
- $- v$ — динозавра номер v съели.
- $? u v$ — у динозавров с номерами u и v возник конфликт и вам надо найти им третейского судью.

Изначально есть один прадинозавр номер 1; гарантируется, что он никогда не будет съеден.

Формат выходных данных

Для каждого запроса типа "?" в выходной файл нужно вывести на отдельной строке одно число — номер самого молодого динозавра, который может выступить в роли третейского судьи.

Пример

| тест | ответ |
|-------|-------|
| 11 | 1 |
| + 1 | 1 |
| + 1 | 2 |
| + 2 | 2 |
| ? 2 3 | 5 |
| ? 1 3 | |
| ? 2 4 | |
| + 4 | |
| + 4 | |
| - 4 | |
| ? 5 6 | |
| ? 5 5 | |

Задача Е. Остовное дерево (70 баллов)

Ограничение по времени: 2.5 секунд
Ограничение по памяти: 64Mb

Дан связный граф G с множеством вершин V и множеством рёбер E . На каждом ребре графа написан положительный целочисленный вес. Остовным деревом T графа G называется такой граф, что его множество вершин V_T ровно совпадает с множеством V , множество его рёбер E_T является подмножеством E и что T является при этом деревом. Требуется найти в G остовное дерево T , у которого наибольший из весов всех рёбер — наименьший возможный среди всех остовных деревьев графа G .

Вершины графа G пронумерованы числами от 1 до n .

Сложность алгоритма должна получиться $O(m + n \log m)$ в худшем случае.

Формат входных данных

В первой строке входа даны числа n и m — количество вершин и рёбер графа соответственно. $2 \leq n \leq 10\,000, 1 \leq m \leq 1\,000\,000$. В каждой из следующих m строк — по три числа a, b и c , означающих, что между вершинами a и b в графе есть ребро веса c . Числа a и b могут совпадать. В этом случае в графе G есть петля. Одно и то же ребро (a, b) может быть указано в файле несколько раз, при этом, возможно, с разными весами. В этом случае считается, что в графе G присутствуют все указанные рёбра в указанном количестве, т.е. есть кратные рёбра. Веса рёбер не превосходят $1\,000\,000\,000$.

Формат выходных данных

Выведите одно число — наибольший вес ребра в искомом остовном дереве T .

Пример

| тест | ответ |
|-----------------------------------------|-------|
| 4 4 1 2 1 2 3 2 3 4 3 4 1 4 | 3 |

Задача F. Секретные агенты (60 баллов)

Ограничение по времени: 2.5 секунд
Ограничение по памяти: 64Mb

Несколько секретных агентов ловят преступника и патрулируют местность, в которой он предположительно находится. Чтобы общаться между собой, они используют рации, передающие сигнал на ограниченное расстояние. Все рации одинаковые, и у них есть максимальное расстояние, на которое можно передать сигнал — R . Агенты весьма ленивы, поэтому они стоят на своих точках и не двигаются.

Как только один из агентов заметит преступника, он должен тут же передать эту новость вместе с координатами преступника всем остальным агентам. Для этого он может передать новость некоторым из тех агентов, до которых добивает его рация, те — расскажут своим соседям и т.д., но ни один агент при этом не должен сойти с места. Необходимо выяснить, какое минимальное значение R устраивает агентов, если задано их изначальное положение.

Формат входных данных

В первой строке входа задано число $1 \leq n \leq 1\,000$ — количество агентов. В каждой из следующих n строк — по два целых числа x и y — координаты соответствующего агента ($-10^9 \leq x, y \leq 10^9$).

Формат выходных данных

В единственную строку выхода необходимо вывести одно вещественное число с десятью знаками после десятичной точки — минимальное возможное значение R . Ваш ответ должен отличаться от правильного не более, чем на 10^{-4} . Мы предлагаем вам вывести десять знаков после десятичной точки, чтобы избежать ошибок округления.

Пример

| тест | ответ |
|-------------------------------|--------------|
| 4 0 0 2 0 2 2 0 2 | 2.0000000000 |