

Задача А. Нефтяное дело

Имя входного файла: oil.in
Имя выходного файла: oil.out
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Вам дан неориентированный связный граф из n вершин и m рёбер. Для каждого ребра известна стоимость его удаления в тугриках. У вас есть s тугриков. Вы хотите удалить как можно больше рёбер так, чтобы граф оставался связным, а суммарная стоимость всех удалённых рёбер не превосходила s тугриков.

Формат входных данных

Первая строка входных данных содержит три целых числа n , m и s — количество вершин в графе, количество рёбер в графе и количество тугриков соответственно ($2 \leq n \leq 50\,000$, $1 \leq m \leq 100\,000$, $0 \leq s \leq 10^{18}$). Следующие m строк содержат описания рёбер графа. Каждое описание состоит из трёх целых чисел — номера вершин, которые соединяет данное ребро, и стоимость удаления ребра в тугриках (стоимость не превосходит 10^9). В графе не бывает кратных рёбер и петель.

Формат выходных данных

В первой строке выведите максимальное количество удаляемых рёбер. Во второй строке выведите номера удаляемых рёбер (рёбра нумеруются с единицы в порядке, данном во входном файле).

Примеры

oil.in	oil.out
6 7 10	2
1 2 3	1 6
1 3 3	
2 3 3	
3 4 1	
4 5 5	
5 6 4	
4 6 5	

Задача В. Дороги короля

Имя входного файла: `roads.in`
Имя выходного файла: `roads.out`
Ограничение по времени: 5 секунд
Ограничение по памяти: 256 мегабайт

Король Байтазар хочет построить новую сеть дорог в Байтландии, состоящей из n городов. В i -м городе проживает a_i людей.

Изначально в королевстве нет ни одной дороги. Король хочет построить несколько дорог таким образом, чтобы из любого города можно было добраться до любого другого. Также, он хочет минимизировать стоимость постройки этих дорог.

Стоимость дороги, соединяющей города i и j равна $(a_i + a_j) \bmod M$ (если хотите мотивацию такой стоимости, прочитайте английскую версию условия).

Помогите королю Байтазару найти минимальную возможную стоимость постройки такой сети дорог.

Формат входных данных

На первой строке находятся два целых числа n и M ($1 \leq n \leq 200\,000$, $1 \leq M \leq 10^9$).

На второй строке находятся n целых чисел a_i , разделенных пробелом, ($0 \leq a_i < M$).

Формат выходных данных

Выведите единственное целое число — минимальную стоимость постройки сети дорог.

Примеры

<code>roads.in</code>	<code>roads.out</code>
5 9 1 3 5 8 8	6

Задача С. Алгоритм двух китайцев

Имя входного файла: `chinese.in`
Имя выходного файла: `chinese.out`
Ограничение по времени: 4 секунды
Ограничение по памяти: 256 мегабайт

Вам дан взвешенный ориентированный граф. Найдите минимальную возможную сумму весов ребер, которых необходимо оставить в графе, чтобы из вершины с номером 1 по этим ребрам можно было добраться до любой другой вершины.

Формат входных данных

В первой строке даны два целых числа n и m ($1 \leq n \leq 1000, 0 \leq m \leq 10000$) — количество вершин и ребер в графе.

В следующих m строках даны ребра графа. Ребро описывается тройкой чисел a_i, b_i и w_i ($1 \leq a_i, b_i \leq n; -10^9 \leq w_i \leq 10^9$) — номер вершины, из которой исходит ребро, номер вершины, в которую входит ребро, и вес ребра.

Формат выходных данных

Если нельзя оставить подмножество ребер так, чтобы из вершины с номером 1 можно было добраться до любой другой, в единственной строке выведите «NO».

Иначе, в первой строке выведите «YES», а во второй строке выведите минимальную возможную сумму весов ребер, которых необходимо оставить.

Примеры

<code>chinese.in</code>	<code>chinese.out</code>
2 1 2 1 10	NO
4 5 1 2 2 1 3 3 1 4 3 2 3 2 2 4 2	YES 6

Задача D. Почтовая реформа

Имя входного файла:	mail.in
Имя выходного файла:	mail.out
Ограничение по времени:	4 секунды
Ограничение по памяти:	256 мегабайт

В Флатландии идет пора реформ. Недавно была проведена реформа дорог, так что теперь по дорогам страны из любого города можно добраться в любой другой, причем только одним способом. Также была проведена реформа волшебников, так что в каждом городе остался ровно один волшебник. Теперь же началась реформа почтовой системы.

Недавно образованное почтовое агентство «Экс-Федя» предлагает уникальную услугу — коллективную посылку. Эта услуга позволяет отправлять посылки жителям всех городов на каком-либо пути по цене обычной посылки. Удивительно, но пользоваться такой услугой стали только волшебники Флатландии, которые стали в большом количестве отправлять друг другу магические кактусы. Агентство столкнулось с непредвиденной проблемой: как известно, все волшебники живут в башнях и мало того, что не строят в них лестницы, так еще время от времени меняют их высоту. Поэтому, чтобы доставить посылку волшебнику, который живет в башне высотой h , курьеру агентства требуется иметь с собой не менее h метров веревки.

Вам поручено руководить отделом логистики — по имеющимся данным о высотах башен и об их изменениях вам нужно определять минимальную длину веревки, которую нужно выдать курьеру, который доставляет посылки между городами i и j .

Формат входных данных

Первая строка входного файла содержит число n — количество городов в Флатландии ($1 \leq n \leq 50\,000$). Во второй строке находится n положительных чисел, не превосходящих 10^5 — высоты башен в городах. В следующих $n - 1$ строках содержится по два числа u_i и v_i — описание i -й дороги, $1 \leq u_i, v_i \leq n, u_i \neq v_i$. В следующей строке содержится число k — количество запросов ($1 \leq k \leq 100\,000$). В следующих k строках содержатся описания запросов в следующем формате:

- Уведомление от волшебника из города i о том, что высота его башни стала равна h , имеет вид $! i h, 1 \leq i \leq n, 1 \leq h \leq 10^5$.
- Запрос от курьера о выдаче веревки для доставки посылок во все города на пути от i до j включительно имеет вид $? i j, 1 \leq i, j \leq n$.

Формат выходных данных

Для каждого запроса доставки посылок выведите минимальную длину веревки, которую необходимо выдать курьеру.

Примеры

mail.in	mail.out
3 1 2 3 1 3 2 3 5 ? 1 2 ! 1 5 ? 2 3 ! 3 2 ? 1 2	3 3 5
1 100 5 ! 1 1 ? 1 1 ! 1 1000 ? 1 1 ! 1 1	1 1000

Задача E. Центроиды дерева

Имя входного файла: centroid.in
Имя выходного файла: centroid.out
Ограничение по времени: 5 секунд
Ограничение по памяти: 256 мегабайт

Дано дерево из n вершин. У каждой вершины есть цвет. Нужно обработать q запросов (v_i, c_i) : найти расстояние от v_i до ближайшей к v_i вершины цвета c_i . Расстоянием между вершинами называется минимальное количество рёбер в пути между ними.

Формат входных данных

На первой строке число n ($1 \leq n \leq 10^5$), следующая строка содержит числа p_1, p_2, \dots, p_{n-1} . $0 \leq p_i < i$. p_i – отец вершины i в дереве. Далее строка с числами a_0, a_1, \dots, a_{n-1} . $0 \leq a_i < n$. a_i – цвет вершины i . Далее строка с числом q ($1 \leq q \leq 10^5$). Следующие q строк содержат запросы $v_i q_i$ ($0 \leq v_i < n, 0 \leq c_i < n$).

Формат выходных данных

Для каждого запроса выведите одно число – расстояние до ближайшей вершины нужного цвета, или -1 , если в дереве нет вершин такого цвета.

Примеры

centroid.in	centroid.out
5	0 1 2 -1 2 1 2 1 1
0 1 1 3	
1 2 3 2 1	
9	
0 1	
0 2	
0 3	
1 0	
2 1	
2 2	
3 3	
3 1	
4 2	

Задача F. На далекой Амазонке

Имя входного файла:	treeeg.in
Имя выходного файла:	treeeg.out
Ограничение по времени:	6.5 секунд
Ограничение по памяти:	256 мегабайт

В бассейне далёкой реки Амазонки расположены N городов, пронумерованных для удобства целыми числами от 1 до N . Всем известно, что местные леса непроходимы, и передвижение возможно только по рекам. Как следствие, схема соединения городов является деревом.

К несчастью, в этом году в бассейне далёкой Амазонки не на шутку разошлась эпидемия новой болезни — крабового гриппа. То и дело поступает информация о новых заболевших. Поначалу справиться с ней было легко, но вскоре почти все больницы были переполнены, и сейчас пациентов может принимать только госпиталь, находящийся в городе 1.

Для удобства граждан была открыта горячая линия, куда первым делом необходимо обратиться при появлении симптомов крабового (его ещё часто называют раковым) гриппа. Вам необходимо написать программу, которая будет отвечать на обращения пострадавших, учитывая при этом информацию о работающих больницах. Вам ещё повезло, что вы знаете все запросы заранее!

Более формально, поступают запросы трёх видов:

- «+ v » — госпиталь города v снова может принимать больных. Гарантируется, что в момент перед этим запросом госпиталь города v не работал.
- «- v » — госпиталь города v не может больше принимать больных. Гарантируется, что в момент перед этим запросом госпиталь города v работал.
- «? v » — заболел человек в городе v , необходимо сообщить ему расстояние до ближайшего города с работающим госпиталем (в идеале неплохо бы ещё и сказать номер этого города, но этим пусть занимаются ваши коллеги). Гарантируется, что в момент такого запроса имеется хотя бы один работающий госпиталь.

Формат входных данных

В первой строке находится единственное число N — количество городов ($1 \leq N \leq 300\,000$). Следующие $N - 1$ строк содержат информацию о соединениях между городами в формате « $u\ v\ l$ », что означает соединение между городами u и v длиной l километров ($1 \leq u, v \leq N$, $1 \leq l \leq 1000$). Направлением течения можно пренебречь и считать, что время движения зависит только от расстояний.

Далее на отдельной строке записано число Q — количество запросов. Следующие Q строк содержат описание запросов в формате « $c\ v$ », где c — это один из трёх символов «+», «-» и «?», а v — номер города ($1 \leq v \leq N$).

Формат выходных данных

Для каждого запроса вида «? v » выведите на отдельной строке одно число — расстояние в километрах до ближайшего города с работающим госпиталем.

Примеры

treeeg.in	treeeg.out
5	6
1 2 2	4
2 3 3	7
3 4 1	
3 5 4	
5	
? 4	
+ 5	
? 3	
- 1	
? 2	