

## Задача А. Чип и Дейл в лабиринте

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

Чип и Дейл спешат на помощь! Но внимательные зрители знают, что помощь как правило нужна самим Чипу и Дейлу, поэтому сегодня вам надо будет сыграть роль сообразительной Гаечки. Итак, Чип и Дейл снова попали в лапы к Толстопузу. Кот очень не любит грызунов и поэтому приготовил им изощренное испытание. Он собирается поместить их в лабиринт и посмотреть смогут ли они из него выбраться. Лабиринт представляет собой дерево, в котором каждое ребро имеет одно направление. Гаечка подслушала разговор Толстопузу со своими сообщниками и теперь знает несколько возможных вариантов: в какую точку лабиринта поместят её друзей, и где будет выход. Для каждого такого варианта она хочет понять, смогут ли Чип и Дейл найти выход, или нет.

### Формат входных данных

В первой строке входного файла записано число  $n$  ( $1 \leq n \leq 10^5$ ) — число вершин в дереве. В следующих  $n - 1$  строках описаны ребра дерева. В  $i + 1$  строке файла записаны два числа  $a_i, b_i$  ( $1 \leq a_i, b_i \leq n$ ), означающие, что существует ребро из  $a_i$  в  $b_i$ .

Далее записано число  $m$  ( $1 \leq m \leq 10^5$ ) — число запросов. После этого идет описание запросов, каждый запрос в новой строке. Для каждого запроса задается  $x_i, y_i$  ( $1 \leq x_i, y_i \leq n$ ) — точка, в которую поместят Чипа и Дейла, и выход из лабиринта соответственно.

### Формат выходных данных

Для каждого запроса надо в отдельной строке вывести Yes, если бурундуки смогут найти выход, и No иначе.

### Пример

стандартный ввод	стандартный вывод
4	Yes
1 2	Yes
3 1	No
4 1	Yes
6	No
1 2	No
3 2	
2 3	
4 2	
4 3	
2 1	

## Задача В. Дуумвират 2

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

Надо бы всё-таки написать нормальную легенду, а то как-то не очень. И без легенды непонятно, почему задача так называется

Но пока легенды нет, вот формальное условие:

Вам дано дерево на  $n$  вершинах. В вершинах записаны числа. Требуется отвечать на запросы двух видов:

- $? v u$  — узнать сумму значений чисел, записанных в вершинах на пути из  $v$  в  $u$ .
- $! v x$  — сделать значение, записанное в вершине  $v$  равным  $x$ .

### Формат входных данных

В первой строке записано число  $n$  — количество вершин дерева ( $1 \leq n \leq 10^5$ ). Во второй строке записаны через пробел  $n$  чисел  $v_i$  ( $|v_i| < 10^9$ ), задающие значения в вершинах. В следующих  $n - 1$  строках описаны ребра дерева. В  $(i + 2)$ -й строке записаны номера вершин  $a_i, b_i$  ( $1 \leq a_i, b_i \leq n$ ), означающие, что в дереве есть ребро из вершины  $a_i$  в вершину  $b_i$ .

Далее на отдельной строке записано число  $m$  — количество запросов ( $1 \leq m \leq 10^5$ ). После этого идут  $m$  строк с описанием запросов, в очередной строке может быть написано  $?vu$  — узнать сумму на пути из  $v$  в  $u$  ( $1 \leq v, u \leq n$ ). Или  $!vx$  — изменить значение в вершине  $v$  на  $x$  ( $1 \leq v \leq n, -10^9 \leq x \leq 10^9$ ).

### Формат выходных данных

Для каждого запроса в виде  $?vu$  выведите искомую величину.

### Пример

стандартный ввод	стандартный вывод
7	0
0 -7 -2 -7 1 4 8	-5
3 6	-2
7 6	-8
1 3	
5 1	
4 6	
2 1	
7	
? 1 1	
? 2 6	
! 2 -8	
! 7 -6	
! 4 -6	
? 1 3	
? 1 2	

## Задача С. Дуумвират

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

Вам дано дерево. В вершинах записаны числа. Нужно научиться находить сумму чисел на пути из  $v$  в  $u$ .

### Формат входных данных

В первой строке записано число  $n$  — количество вершин дерева ( $1 \leq n \leq 10^5$ ). Во второй строке записаны через пробел  $n$  чисел  $v_i$  ( $|v_i| < 10^9$ ), задающие значения в вершинах. В следующих  $n - 1$  строках описаны ребра дерева. В  $(i + 2)$ -й строке записаны номера вершин  $a_i, b_i$  ( $1 \leq a_i, b_i \leq n$ ), означающие, что в дереве есть ребро из вершины  $a_i$  в вершину  $b_i$ .

Далее на отдельной строке записано число  $m$  — количество запросов ( $1 \leq m \leq 10^5$ ). После этого идут  $m$  строк с описанием запросов, в  $(n + 2 + i)$ -й строке записаны через пробел числа  $x_i$  и  $y_i$  ( $1 \leq x_i, y_i \leq n$ ).

### Формат выходных данных

Для каждого запроса на отдельной строке требуется вывести сумму всех значений  $v_i$  по всем вершинам на пути из  $x_i$  в  $y_i$ .

### Пример

стандартный ввод	стандартный вывод
4	-15
-9 -6 -1 9	-16
1 2	-16
3 1	-6
4 1	-1
6	-15
1 2	
3 2	
2 3	
4 2	
4 3	
2 1	

## Задача D. Разреженные таблицы

Имя входного файла: `sparse.in`  
 Имя выходного файла: `sparse.out`  
 Ограничение по времени: 2 секунды  
 Ограничение по памяти: 64 мегабайта

64 мегабайта

Дан массив из  $n$  чисел. Требуется написать программу, которая будет отвечать на запросы следующего вида: найти минимум на отрезке между  $u$  и  $v$  включительно.

### Формат входных данных

В первой строке входного файла даны три натуральных числа  $n$ ,  $m$  ( $1 \leq n \leq 10^5$ ,  $1 \leq m \leq 10^7$ ) и  $a_1$  ( $0 \leq a_1 < 16\,714\,589$ ) — количество элементов в массиве, количество запросов и первый элемент массива соответственно. Вторая строка содержит два натуральных числа  $u_1$  и  $v_1$  ( $1 \leq u_1, v_1 \leq n$ ) — первый запрос.

Элементы  $a_2, a_3, \dots, a_n$  задаются следующей формулой:

$$a_{i+1} = (23 \cdot a_i + 21563) \bmod 16714589.$$

Например, при  $n = 10$ ,  $a_1 = 12345$  получается следующий массив:  $a = (12345, 305498, 7048017, 11694653, 1565158, 2591019, 9471233, 570265, 13137658, 1325095)$ .

Запросы генерируются следующим образом:

$$\begin{aligned} u_{i+1} &= ((17 \cdot u_i + 751 + ans_i + 2i) \bmod n) + 1, \\ v_{i+1} &= ((13 \cdot v_i + 593 + ans_i + 5i) \bmod n) + 1, \end{aligned}$$

где  $ans_i$  — ответ на запрос номер  $i$ .

Обратите внимание, что  $u_i$  может быть больше, чем  $v_i$ .

### Формат выходных данных

В выходной файл выведите  $u_m$ ,  $v_m$  и  $ans_m$  (последний запрос и ответ на него).

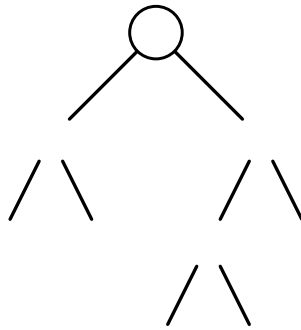
### Пример

<code>sparse.in</code>	<code>sparse.out</code>
10 8 12345 3 9	5 3 1565158

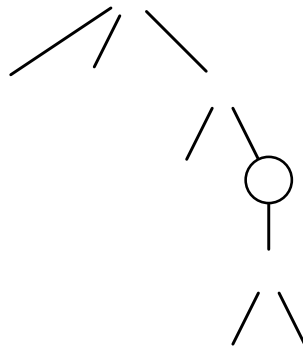
## Задача E. Dynamic LCA

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

Постановка задачи о *наименьшем общем предке* такова: дано дерево  $T$  с выделенным корнем и две вершины  $u$  и  $v$ ,  $\text{lca}(u, v)$  — вершина с максимальной глубиной, которая является предком и  $u$ , и  $v$ . Например, на картинке внизу  $\text{lca}(8, 7)$  — вершина 3.



С помощью операции  $\text{chroot}(u)$  мы можем менять корень дерева, достаточно отметить  $u$ , как новый корень, и направить ребра вдоль пути от корня. Наименьшие общие предки вершин поменяются соответственно. Например, если мы сделаем  $\text{chroot}(6)$  на картинке сверху,  $\text{lca}(8, 7)$  станет вершина 6. Получившееся дерево изображено внизу.



Вам дано дерево  $T$ . Изначально корень этого дерева — вершина 1. Напишите программу, которая поддерживает эти две операции:  $\text{lca}(u, v)$  и  $\text{chroot}(u)$ .

### Формат входных данных

Входной файл состоит из нескольких тестов.

Первая строка каждого теста содержит натуральное число  $n$  — количество вершин в дереве ( $1 \leq n \leq 100\,000$ ). Следующие  $n - 1$  строк содержат по 2 натуральных числа и описывают ребра дерева. Далее идет строка с единственным натуральным числом  $m$  — число операций. Следующие  $m$  строк содержат операции. Строка  $? u v$  означает операцию  $\text{lca}(u, v)$ , а строка  $! u$  —  $\text{chroot}(u)$ . Последняя строка содержит число 0.

Сумма  $n$  для всех тестов не превосходит 100 000. Сумма  $m$  для всех тестов не превосходит 200 000.

### Формат выходных данных

Для каждой операции  $? u v$  выведите значение  $\text{lca}(u, v)$ . Числа разделяйте переводами строк.

**Пример**

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

## Задача F. Праздник к нам приходит

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

В уездном городе Т все заняты подготовкой к новому году. Таня вот, например, готовится вручать подарки в новогоднюю ночь (в городе Т именно Таня исполняет роль Деда Мороза).

В городе Т  $n$  домов, некоторые из которых соединены улицами, причем так, что между любыми двумя домами есть ровно один путь.

Таня уже заготовила  $m$  мешков с подарками (в этом году она дарит серые футболки с желтым единорогом), но столкнулась с проблемой. Ей ведь самой тоже нужно найти дом в городе Т для празднования Нового Года. Раздачей подарков же, как обычно, будут заниматься олениа. Одному олененку можно дать ровно один мешок и отправить его в путь. При этом олениа не ходят по одной и той же улице дважды.  $i$ -й мешок предназначен для жителей всех домов на пути от  $a_i$  до  $b_i$ . Поэтому считается, что олененку можно дать  $i$ -й мешок с подарками, если он сможет выйти из места празднования Тани и пройти через все дома на пути от  $a_i$  до  $b_i$ , при этом не проходя через одну и ту же улицу дважды.

Помогите Тани найти дом для празднования так, чтобы она смогла отправить как можно больше мешков с подарками. В данной задаче можно считать, что количество олениа не ограничено.

### Формат входных данных

В первой строке задано число  $n$  — количество домов в городе Т ( $2 \leq n \leq 2 \cdot 10^5$ ).

В следующих  $n - 1$  строках описаны улицы. Улица задаётся числами  $x_i$  и  $y_i$  — номерами домов, которые она соединяет ( $1 \leq x_i, y_i \leq n, x_i \neq y_i$ ). Гарантируется, что между любыми двумя домами существует единственный путь.

В следующей строке задано число  $m$  — количество мешков у Тани ( $1 \leq m \leq 2 \cdot 10^5$ ).

В следующих  $m$  строках описаны мешки с футболками. В  $i$ -й из них заданы числа  $a_i$  и  $b_i$  — начало и конец  $i$ -го пути ( $1 \leq a_i, b_i \leq n, a_i \neq b_i$ ). Пути могут пересекаться и совпадать.

Дома нумеруются с единицы.

### Формат выходных данных

Выведите одно число — максимальное число мешков с подарками, которые Таня сможет раздать, если выберет оптимальный дом для празднования.

### Пример

стандартный ввод	стандартный вывод
7	2
1 2	
2 3	
3 4	
3 5	
5 6	
5 7	
3	
1 5	
2 4	
6 7	

## Задача G. LCA Problem

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	5 секунд
Ограничение по памяти:	256 мегабайт

Задано подвешенное дерево, содержащее  $n$  ( $1 \leq n \leq 10^5$ ) вершин, пронумерованных от 0 до  $n-1$ . Требуется ответить на  $m$  ( $1 \leq m \leq 10^6$ ) запросов о наименьшем общем предке для пары вершин.

Запросы генерируются следующим образом. Заданы числа  $a_1, a_2$  и числа  $x, y, z$ . Числа  $a_3, \dots, a_{2m}$  генерируются следующим образом:  $a_i = (x \cdot a_{i-2} + y \cdot a_{i-1} + z) \bmod n$ . Первый запрос имеет вид  $(a_1, a_2)$ . Если ответ на  $i-1$ -й запрос равен  $v$ , то  $i$ -й запрос имеет вид  $((a_{2i-1} + v) \bmod n, a_{2i})$ .

### Формат входных данных

Первая строка содержит два числа:  $n$  и  $m$ . Корень дерева имеет номер 0.

Вторая строка содержит  $n-1$  целых чисел,  $i$ -е из этих чисел равно номеру родителя вершины  $i$ .

Третья строка содержит два целых числа в диапазоне от 0 до  $n-1$ :  $a_1$  и  $a_2$ .

Четвертая строка содержит три целых числа:  $x, y, z$ , эти числа неотрицательны и не превосходят  $10^9$ .

### Формат выходных данных

Выведите в выходной файл сумму номеров вершин — ответов на все запросы.

### Примеры

стандартный ввод	стандартный вывод
3 2 0 1 2 1 1 1 0	2
1 2 0 0 1 1 1	0



## Задача Н. Зигмунд Фрейд и Карл Юнг

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

Известный психиатр Зигмунд Фрейд в своей книге "Толкование сновидений" подробно описал, что ему снится, когда в его генеологическое дерево добавляется новый лист. В своём более позднем труде "Я и оно" он также описал ощущения человека, видевшего сон про удаление вершины из генеологического дерева. Несколькими годами позже молодой Карл Юнг — будущий не менее известный психиатр, изучая работы своего знаменитого предшественника, не мог пройти мимо тех работ и стал готовить грандиозный эксперимент, основанный на строго задокументированных показаниях о более, чем ста тысячах опрошенных. Для завершения эксперимента не хватает совсем немногого — быстро находить наименьшего общего предка двух вершин.

Несмотря на то, что Юнг при жизни так и не закончил эксперимент, мы уверены, что он будет Вам безмерно благодарен, если Вы довершите его гениальную задумку.

### Формат входных данных

Во входном файле записано число  $q$ , обозначающее количество запросов ( $1 \leq q \leq 2 \cdot 10^5$ ). Далее на отдельных строках следуют  $q$  запросов, обозначающих следующие события:

- $+ v$  — добавился новый лист, его предком стала вершина с номером  $v$ . Добавившейся вершине нужно присвоить наименьший натуральный номер, который до этого еще никогда не встречался.
- $- v$  — вершина с номером  $v$  удалилась из дерева, предком её детей становится её предок.
- $? u v$  — Карл Юнг интересуется наименьшим общим предком вершин  $u$  и  $v$ .

Изначально есть одна вершина с номером 1, гарантируется, что она никогда не будет удалена

### Формат выходных данных

Для каждого запроса типа «?» в выходной файл нужно вывести на отдельной строке одно число — номер вершины интересующей Юнга

### Пример

стандартный ввод	стандартный вывод
11	1
+ 1	1
+ 1	2
+ 2	2
? 2 3	5
? 1 3	
? 2 4	
+ 4	
+ 4	
- 4	
? 5 6	
? 5 5	

## Задача I. LCA Problem Revisited

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	5 секунд
Ограничение по памяти:	256 мегабайт

Задано подвешенное дерево, содержащее  $n$  ( $1 \leq n \leq 10^5$ ) вершин, пронумерованных от 0 до  $n-1$ . Требуется ответить на  $m$  ( $1 \leq m \leq 10^7$ ) запросов о наименьшем общем предке для пары вершин.

Запросы генерируются следующим образом. Заданы числа  $a_1, a_2$  и числа  $x, y, z$ . Числа  $a_3, \dots, a_{2m}$  генерируются следующим образом:  $a_i = (x \cdot a_{i-2} + y \cdot a_{i-1} + z) \bmod n$ . Первый запрос имеет вид  $(a_1, a_2)$ . Если ответ на  $i-1$ -й запрос равен  $v$ , то  $i$ -й запрос имеет вид  $((a_{2i-1} + v) \bmod n, a_{2i})$ .

### Формат входных данных

Первая строка содержит два числа:  $n$  и  $m$ . Корень дерева имеет номер 0.

Вторая строка содержит  $n-1$  целых чисел,  $i$ -е из этих чисел равно номеру родителя вершины  $i$ .

Третья строка содержит два целых числа в диапазоне от 0 до  $n-1$ :  $a_1$  и  $a_2$ .

Четвертая строка содержит три целых числа:  $x, y, z$ , эти числа неотрицательны и не превосходят  $10^9$ .

### Формат выходных данных

Выведите в выходной файл сумму номеров вершин — ответов на все запросы.

### Примеры

стандартный ввод	стандартный вывод
3 2 0 1 2 1 1 1 0	2
1 2 0 0 1 1 1	0

## Задача J. Гремучая ива

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

Гремучей Иве  $n$  лет. На первом году жизни у неё был только \*корень\*. Далее каждый год Ива отращивала себе \*ветку\* длиной ровно один фут, которая оканчивается \*листом\*. Корень тоже считается листом.

Гарри Поттер и Рон Уизли очень часто нужно проникать в Визжащую Хижину, которую охраняет Ива. Они решили подойти основательно и тщательно исследовать строение Гремучих Ив. Они просят вас найти историю изменения \*диаметров\* Ивы. Диаметр Ивы называется максимальное расстояние в футах между двумя листьями дерева.

### Формат входных данных

Первая строке содержит целое число  $n$  — возраст ивы ( $1 \leq n \leq 10^6$ ).

Следующие  $n$  строк содержат описание её листьев. Каждая строка содержит номер листа  $p_i$ , который является её родителем ( $1 \leq p_i \leq i$ ). Корень имеет номер 1.

### Формат выходных данных

Выведите диаметр после добавления каждого листа.

### Примеры

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

### Замечание

Задача стоит **7 баллов**.