

## Задача А. Камни HARD

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

Гора состоит из площадок, соединенных между собой узкими проходами. На каждой площадке лежит какое-то количество камней. У разных площадок разный рейтинг, зависящий от высоты площадки — чем выше площадка, тем больше её рейтинг среди камней. Очевидно, что на площадках с более высоким рейтингом лежит большее количество камней.

В один солнечный день главный камень одной из площадок обратил внимание, что вокруг стало очень тесно. Но камни не могут катиться вверх, а на нижние площадки они катиться не хотят из-за плохого рейтинга. Поэтому главный камень пошел на хитрость. Он выбрал  $M$  камней и сказал каждому из них, что где-то внизу появилась замечательная площадка с самым высоким рейтингом на горе. Для полной правдоподобности он составил карту площадок, на которые можно скатиться с его площадки. Всего (включая переполненную площадку) оказалось  $N$  площадок. Поскольку главный камень ленивый, он не стал рисовать все пути между площадками, а просто нарисовал минимальное количество проходов между площадками, чтобы из его площадки можно было добраться в любую, находящуюся ниже. Он показал каждому из  $M$  камней эту карту и объяснил, где находится замечательная площадка (чтобы камень запомнил путь). Причем, чтобы главные камни нижних площадок не имели к нему претензий из-за такого нашествия, некоторым камням он показал на другие площадки, чем остальным. Чтобы не создать лавину, он сказал камням катиться с небольшим интервалом между собой. Немного подумав, он решил, что можно пускать камни парами, чтобы пространство быстрее освободилось. Камни ему поверили и стали собираться в дорогу.

Камни не любят скучать, поэтому каждая пара решила посчитать, сколько времени они будут катиться вместе, и, может быть, перестроить пары, чтобы увеличить это время. На то, чтобы скатиться по проходу с одной площадки в следующую, не заходя по пути на другие площадки, камень тратит одну минуту. Но складывать числа камни не умеют, в этом им требуется ваша помощь.

### Формат входных данных

В первой строке входного файла записаны два целых числа —  $N$  и  $K$ , где  $1 \leq N \leq 500\,000$ ,  $K = M/2$ ,  $0 \leq M \leq 1000\,000$ .

В следующих  $(N - 1)$  строках находятся пары чисел  $i$  и  $j$ , означающие, что из  $i$ -й площадки можно напрямую скатиться на  $j$ -ю. Площадки занумерованы числами от 1 до  $N$ , верхняя площадка, откуда катятся камни, может иметь любой номер. Записанных пар номеров площадок достаточно, чтобы определить путь из самой верхней площадки до любой другой.

Дальше идут  $K$  строк, на каждой строке через пробел записана пара чисел  $a$  и  $b$  ( $1 \leq a, b \leq N$ ), обозначающая пункты назначения для очередной пары камней: первый камень катится на площадку  $a$ , а второй — на площадку  $b$ .

### Формат выходных данных

В выходной файл нужно вывести  $K$  строк, на каждой строке должно находиться по три целых числа  $a$ ,  $b$  и  $c$ , записанных через пробел. Эти числа означают, что пара камней, направляющихся на площадки  $a$  и  $b$ , будет катиться вместе  $c$  минут. Тройки чисел нужно выдавать в том же порядке, в котором перечислены соответствующие им пары во входном файле.

## Пример

стандартный ввод	стандартный вывод
6 2	2 4 0
1 2	5 6 1
1 3	
1 4	
3 5	
3 6	
2 4	
5 6	

## Задача В. 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 C. LA Problem Revisited

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 6 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 поиске  $k$ -го предка для вершины.

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 относится к вершине  $a_1$ . Пусть  $d_1$  — глубина вершины  $a_1$  в дереве (корень имеет глубину 1, дети корня — 2, их дети — 3 и т.д.). Тогда в запросе ищется  $(a_2 \bmod d_1)$ -й предок вершины  $a_1$ . Если ответ на  $i-1$ -й запрос равен  $v$ , то  $i$ -й запрос относится к вершине  $a_{2i-1}$ . Пусть  $d_i$  — глубина вершины  $a_{2i-1}$ . Тогда в запросе ищется  $((a_{2i} + v) \bmod d_i)$ -й предок вершины  $a_{2i-1}$ . Нулевым предком вершины считается она сама.

### 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	1
1 2 0 0 1 1 1	0

## Задача D. Dynamic LCA

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 5 секунд  
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  $v$ .

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.

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 Е. 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 F. Цветные волшебники

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

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. Поэтому 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:  $n$  ( $1 \leq n \leq 10^5$ ) и  $m$  ( $1 \leq m \leq 5 \cdot 10^5$ ) — количество городов и дорог в волшебной стране соответственно.

Вторая 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  $k$  ( $1 \leq k \leq 10^5$ ) — 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

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

## Задача G. Простыни

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

В один прекрасный день маленький Дональд решил вымыть  $N$  своих чистых белых простыней. После мытья он положил их на землю во дворе, чтобы их высушить. Дональд помещал простыни так, чтобы никакие две простыни не касались ни сторонами, ни углами, и чтобы их стороны не пересекались, но возможно, что он разместил меньшие простыни поверх более крупных или что простыня полностью закрывает какие-то другие простыни. Сделав это, Дональд лег спать.

Друг Дональда Ким как-то узнал о том, что Дональд сушит простыни и решил пообщаться с ним. Он нашел пейнтбольный пистолет своего отца на чердаке. Наряду с пистолетом у него было несколько пейнтбольных мячей, каждый из них имел свой цвет (не обязательно уникальный). Как только Дональд заснул, Ким вошёл во двор к Дональду и начал стрелять пр простыням из пейнтбольного пистолета. Простыни Дональда очень тонкие, поэтому, когда Ким попадает в какую-то простыню, она пропускает краску дальше, на простыню ниже (и та тоже, и так происходит, пока не закончатся простыни и краска не попадет на землю). После того, как Ким использовал все шары, он с радостью покинул двор Дональда. Дональд был очень расстроен, увидев, что случилось с его простынями. Дональд очень заинтересован в правильных данных о преступлении Кима, но он в шоке и не способен думать, поэтому просит вас сказать ему количество цветов на каждой простыне.

Мы можем представлять двор Дональда как бесконечную систему координат, а простыни - как прямоугольники, параллельные осям координат. Выстрелы Кима могут быть представлены как точки в этой системе.

Когда-то в детстве дедушка рассказал Киму, что снаряд никогда не попадает в одну воронку дважды, так что координаты всех выстрелов попарно различны.

### Формат входных данных

Первая строка ввода содержит два числа - количество простынь  $N$  ( $1 \leq N \leq 300000$ ), и количество шаров  $M$  ( $1 \leq M \leq 300000$ ).

$i$ -я из следующих  $N$  строк содержит четыре числа: координаты нижнего левого угла  $A_i, B_i$  ( $1 \leq A_i, B_i \leq 10^9$ ) и верхнего правого угла  $C_i, D_i$ , ( $1 \leq C_i, D_i \leq 10^9$ )  $i$ -й простыни.

$j$ -я из следующих  $M$  строк содержит три числа, где  $X_j, Y_j$  ( $1 \leq X_j, Y_j \leq 10^9$ ) - координаты  $j$ -го выстрела Кима и  $K_j$  ( $1 \leq K_j \leq 10^9$ ) - цвет  $j$ -го пейнтбольного шара.

### Формат выходных данных

$i$ -я из  $N$  выходных строк должна содержать количество цветов на  $i$ -м листе.



## Примеры

стандартный ввод	стандартный вывод
2 2 1 1 3 3 5 6 10 10 3 3 1 5 1 2	1 0
3 3 1 1 7 7 2 2 6 6 3 3 5 5 4 4 1 2 6 2 4 7 3	3 2 1
1 3 1 1 7 7 2 6 2 4 7 3 4 4 1	3

## Задача Н. Задача, которую никто не сдаст

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 6 секунд  
Ограничение по памяти: 256 мегабайт

К сожалению, у меня не хватило времени, чтобы написать нескучное условие к этой задаче, тем более всё равно её никто не сдаст.

Вам дан массив  $a_0, \dots, a_{n-1}$ , который задаётся числами  $n, a_0, b_0, c, d$  следующим образом:

$$b_i = (b_{i-1} \cdot c + d) \bmod 2^{31} \text{ для } i \geq 1$$

$$a_i = (a_{i-1} + 1 - 2 \cdot ((b_i \bmod 239179) \bmod 2)) \text{ для } i \geq 1$$

Обратите внимание, что два соседних числа отличаются либо на  $+1$ , либо на  $-1$ .

Ответьте на  $n$  запросов,  $i$ -й запрос — максимум на отрезке  $[\min(l_i, r_i), \max(l_i, r_i)]$  для  $i = 0 \dots n-1$ .

Пусть  $ans_i$  — ответ на  $i$ -й запрос. Будем считать, что  $ans_{-1} = 0$ . Вам задаётся число  $x_0$ . Далее,  $l_i$  и  $r_i, x_i$  вычисляются так:

$$l_i = (x_i + ans_{i-1}) \bmod n \text{ для } i \geq 0$$

$$r_i = (l_i + i) \bmod n \text{ для } i \geq 0$$

$$x_i = (x_{i-1} \cdot 1103515245 + 12345) \bmod 2^{31} \text{ для } i \geq 1$$

При взятии по модулю обратите внимание на то, что  $ans_i$  бывают отрицательными.

### Формат входных данных

Единственная строка ввода содержит шесть целых чисел  $n, a_0, b, c, d, x_0$ .

- $1 \leq n \leq 2 \cdot 10^7$
- $-10^9 \leq a_0 \leq 10^9$
- $0 \leq b, c, d, x_0 \leq 2^{31} - 1$

### Формат выходных данных

Выведите сумму ответов на все запросы.

### Примеры

стандартный ввод	стандартный вывод
4 1 2 3 4 0	11
100500 -1 23 45 67 89	-8614564

## Задача I. Задача, которую никто не прочитает

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 6 секунд  
Ограничение по памяти: 512 мегабайт

Вам дан массив  $a_0, \dots, a_{n-1}$ , который задаётся числами  $n, a_0, b_0, c, d$  следующим образом:

$$b_i = (b_{i-1} \cdot c + d) \bmod 2^{31} \text{ для } i \geq 1$$

$$a_i = (a_{i-1} + 1 - 2 \cdot ((b_i \bmod 239179) \bmod 2)) \text{ для } i \geq 1$$

Обратите внимание, что два соседних числа отличаются либо на  $+1$ , либо на  $-1$ .

Ответьте на  $n$  запросов,  $i$ -й запрос — поиск количества максимумов на отрезке  $[\min(l_i, r_i), \max(l_i, r_i)]$  для  $i = 0 \dots n - 1$ . Пусть  $ans_i$  — ответ на  $i$ -й запрос. Будем считать, что  $ans_{-1} = 0$ . Вам задаётся число  $x_0$ . Далее,  $l_i$  и  $r_i, x_i$  вычисляются так:

$$l_i = (x_i + ans_{i-1}) \bmod n \text{ для } i \geq 0$$

$$r_i = (l + i) \bmod n \text{ для } i \geq 0$$

$$x_i = (x_{i-1} \cdot 1103515245 + 12345) \bmod 2^{31} \text{ для } i \geq 1$$

### Формат входных данных

Единственная строка ввода содержит шесть целых чисел  $n, a_0, b, c, d, x_0$ .

- $1 \leq n \leq 2 \cdot 10^7$
- $-10^9 \leq a_0 \leq 10^9$
- $0 \leq b, c, d, x_0 \leq 2^{31} - 1$

### Формат выходных данных

Выведите сумму ответов на все запросы.

### Примеры

стандартный ввод	стандартный вывод
4 1 2 3 4 0	4
100500 -1 23 45 67 89	173287