

Задача А. Сортировка

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

Дана перестановка p длины n и m пар чисел x_i, y_i от 1 до n . Требуется обработать q запросов, каждый запрос это пара чисел $1 \leq a, b \leq n$, при запросе нужно поменять p_a и p_b местами. Требуется определить, можно ли отсортировать перестановку какой-то последовательностью обменов элементов с индексами x_i, y_i после каждого запроса.

Формат входных данных

В первой строке вводятся три числа — n, m, q . Гарантируется, что $1 \leq n, m, q \leq 2 \cdot 10^5$.

Во второй строке содержатся n чисел — p_1, p_2, \dots, p_n . Гарантируется, что p — перестановка длины n .

Следующие m строк содержат пары x_i, y_i . Гарантируется, что $1 \leq x_i, y_i \leq n$. Следующие q строк содержат запросы, то есть пары a_i, b_i . Гарантируется, что $1 \leq a_i, b_i \leq n$.

Формат выходных данных

Введите строку длины q из символов «N» и «Y».

Примеры

стандартный ввод	стандартный вывод
5 3 6 1 4 2 3 5 3 4 1 2 4 1 1 3 1 4 3 4 3 1 3 4 5 5	YYYYYY
5 1 7 1 3 2 4 5 2 3 2 4 1 2 2 3 3 4 4 1 1 2 2 3	NNNNYNN
7 7 10 3 2 1 5 4 6 7 3 1 3 4 2 4 1 4 2 5 5 5 1 2 7 1 4 6 4 6 1 7 2 2 3 7 3 7 5 4 2 6 2 6	NNNYNYNY

Задача В. Взлом ПИН-кода

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

Это интерактивная задача.

ПИН-код для банковской карты задаётся строкой из четырёх цифр. Возможные значения ПИН-кода от «0000» до «9999».

Внимательно ознакомьтесь с документом <https://www.ietf.org/rfc/rfc1321.html>. В нём описан алгоритм вычисления md5 хэша.

Из соображений безопасности ПИН-коды VIP клиентов не хранятся в базе данных явно, вместо этого банк хранит их md5 хэши.

Вашей группировке удалось похитить один такой md5 хэш. Подберите подходящий ПИН-код. У вас будет **три** попытки до того, как карта окажется заблокирована.

Протокол взаимодействия

Вы начинаете взаимодействие со считывания строки длины 32 — валидного md5 хэша загаданного ПИН-кода.

Вы можете не более **трёх** раз ввести на проверку ПИН-код.

Если ПИН-код **верный**, интерактор выведет «ok». После получения такого сообщения вы обязаны немедленно завершить программу.

Если ПИН-код **неверный**, интерактор выведет «wrong». После получения такого сообщения вы можете продолжать проверять ПИН-коды, если у вас ещё остались попытки.

Пример

стандартный ввод	стандартный вывод
312351bff07989769097660a56395065	
wrong	1234
wrong	0000
ok	2025

Задача С. Шифр

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

Вася хочет передать Ане секретное послание M из n букв латинского алфавита. К сожалению, у них нет общего секретного ключа для шифрования, поэтому они решили действовать следующим образом.

Каждый из участников генерирует случайный ключ из n чисел от 0 до 25. Пусть ключ Васи B , а ключ Ани A . Обозначим буквы от A до Z числами от 0 до 25. Вася шифрует сообщение M следующим образом: $P_i = (M_i + B_i) \bmod 26$, и отправляет строку P Ане. Аня, в свою очередь, шифрует полученное сообщение своим ключом: $Q_i = (P_i + A_i) \bmod 26$ и отправляет его обратно Васе. Теперь Вася проводит обратное преобразование: $R_i = (Q_i - B_i) \bmod 26$, и отправляет строку R Ане. Она также проводит обратное преобразование: $M_i = (R_i - A_i) \bmod 26$, получая исходное сообщение.

Майор Иванов очень не любит, когда кто-то что-то скрывает, и очень хочет узнать, что же такое секретное сообщил Ане Вася. Он уже перехватил сообщения P , Q и R и теперь хочет узнать сообщение M . Помогите ему.

Формат входных данных

Входной файл содержит три строки одинаковой длины, состоящие из заглавных букв латинского алфавита. Длина строк не больше 50.

Формат выходных данных

Выведите секретное сообщение

Пример

стандартный ввод	стандартный вывод
LIJEGHIGU YTKQHJIFQ CCJAAGRRG	PRIOZERSK

Задача D. Новогоднее дерево

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

Маленький олененок гулял по лесу и нашел граф. Всего в графе n вершин, пронумерованные от 1 до n . Между каждой парой вершин есть ребро. Из данного графа олененок решил сделать дерево. Чтобы оно было больше похоже на новогоднее, он решил построить на графе минимальное остовное дерево, где вас каждого ребра побитовое **and** чисел вершин ребра.

Побитовое **and** — бинарная операция, действие которой эквивалентно применению логического «И» к каждой паре битов, которые стоят на одинаковых позициях в двоичных представлениях операндов. Например, побитовое **and** 13 и 6 = 4. Данная операция существует во всех современных языках программирования, например, в языках C++, Java и Python она обозначена как «&», в Pascal — как «and».

Формат входных данных

В первой строке вводится целое число n — количество вершин в графе ($1 \leq n \leq 10^5$).

Формат выходных данных

В единственной строке входных данных выведите вес минимального остовного дерева данного графа.

Примеры

стандартный ввод	стандартный вывод
2	0
4	0

Задача Е. Необычное знакомство

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

Саше понравилась девочка. К сожалению, между Сашей и девочкой встала огромная грозная горилла, поэтому он боится подойти и познакомиться. Единственным вариантом остаётся как-то узнать телеграмм этой девочки, чтобы позже связаться с ней.

Саша очень торопится прочитать лекцию, поэтому у него нет времени решать эту задачу. Напишите код, который будет вставлен в начало следующего фрагмента:

```
// YOUR code

class Gorilla {
    // some secret code
};

class Tyan {
    // some secret code
};

const char* get_tg(const Tyan& tyan) {
    return "Meow!";
}
```

Помогите Саше добиться того, чтобы функция `get_tg` возвращала телеграмм переданной девочки.

Замечание

- Подпрограмма `main`, вызывающая функцию `get_tg`, находится в **другой** единице трансляции.
- Ваше решение **не** должно что-то считывать или что-то выводить.

Задача F. Загадочный лес 1

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

Ёжик гулял по загадочному лесу и встретил таинственного деда волшебника. Он предложил ежику пройти пять испытаний. Если он справится, то получит подарок. Ёжик конечно не стал отказываться от такого предложения. Сейчас он проходит первое по сложности (самое простое) испытание.

Волшебник очень таинственный, поэтому он дал Ёжику следующее задание: написать реализацию функции:

```
int solve(int a, int b, int c) {  
    // ...  
}
```

Конечно не любую функцию Ёжик может написать и она должна выводить такой же результат, как и https://ugadalka.algocourses.ru/?task_id=1. Ограничения на входные данные: $1 \leq a, b, c \leq 1000$.

Гарантируется, что существует реализация функции без ветвлений (условного оператора, тернарного оператора и т.д.). Также Дед очень честный и не использовал в функции какие-то странные числа (например 1231, 8190, 52, ...).

Отправьте реализацию функцию `solve` в следующем формате:

```
int solve(int a, int b, int c) {  
    // YOUR CODE  
}
```

Ничего кроме функции `solve` сдавать **не нужно**.

Задача G. Загадочный лес 2

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

Ёжик гулял по загадочному лесу и встретил таинственного деда волшебника. Он предложил ежику пройти пять испытаний. Если он справится, то получит подарок. Ёжик конечно не стал отказываться от такого предложения. Сейчас он проходит второе по сложности испытание.

Волшебник очень таинственный, поэтому он дал Ёжику следующее задание: написать реализацию функции:

```
int solve(int a, int b, int c) {  
    // ...  
}
```

Конечно не любую функцию Ёжик может написать и она должна выводить такой же результат, как и https://ugadalka.algocourses.ru/?task_id=2. Ограничения на входные данные: $1 \leq a, b, c \leq 1000$.

Гарантируется, что существует реализация функции без ветвлений (условного оператора, тернарного оператора и т.д.). Также Дед очень честный и не использовал в функции какие-то странные числа (например 1231, 8190, 52, ...).

Отправьте реализацию функцию `solve` в следующем формате:

```
int solve(int a, int b, int c) {  
    // YOUR CODE  
}
```

Ничего кроме функции `solve` сдавать **не нужно**.

Задача Н. Загадочный лес 3

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

Ёжик гулял по загадочному лесу и встретил таинственного деда волшебника. Он предложил ежику пройти пять испытаний. Если он справится, то получит подарок. Ёжик конечно не стал отказываться от такого предложения. Сейчас он проходит третье по сложности испытание.

Волшебник очень таинственный, поэтому он дал Ёжику следующее задание: написать реализацию функции:

```
int solve(int a, int b, int c) {  
    // ...  
}
```

Конечно не любую функцию Ёжик может написать и она должна выводить такой же результат, как и https://ugadalka.algocourses.ru/?task_id=3. Ограничения на входные данные: $1 \leq a, b, c \leq 1000$.

Гарантируется, что существует реализация функции без ветвлений (условного оператора, тернарного оператора и т.д.). Также Дед очень честный и не использовал в функции какие-то странные числа (например 1231, 8190, 52, ...).

Отправьте реализацию функцию `solve` в следующем формате:

```
int solve(int a, int b, int c) {  
    // YOUR CODE  
}
```

Ничего кроме функции `solve` сдавать **не нужно**.

Задача I. Загадочный лес 4

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

Ёжик гулял по загадочному лесу и встретил таинственного деда волшебника. Он предложил ежику пройти пять испытаний. Если он справится, то получит подарок. Ёжик конечно не стал отказываться от такого предложения. Сейчас он проходит четвертое по сложности испытание.

Волшебник очень таинственный, поэтому он дал Ёжику следующее задание: написать реализацию функции:

```
int solve(int a, int b, int c) {  
    // ...  
}
```

Конечно не любую функцию Ёжик может написать и она должна выводить такой же результат, как и https://ugadalka.algocourses.ru/?task_id=4. Ограничения на входные данные: $1 \leq a, b, c \leq 1000$.

Гарантируется, что существует реализация функции без ветвлений (условного оператора, тернарного оператора и т.д.). Также Дед очень честный и не использовал в функции какие-то странные числа (например 1231, 8190, 52, ...).

Отправьте реализацию функцию `solve` в следующем формате:

```
int solve(int a, int b, int c) {  
    // YOUR CODE  
}
```

Ничего кроме функции `solve` сдавать **не нужно**.

Задача J. Загадочный лес 5

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

Ёжик гулял по загадочному лесу и встретил таинственного деда волшебника. Он предложил ежику пройти пять испытаний. Если он справится, то получит подарок. Ёжик конечно не стал отказываться от такого предложения. Сейчас он проходит пятое по сложности (самое сложное) испытание.

Волшебник очень таинственный, поэтому он дал Ёжику следующее задание: написать реализацию функции:

```
int solve(int a, int b, int c) {  
    // ...  
}
```

Конечно не любую функцию Ёжик может написать и она должна выводить такой же результат, как и https://ugadalka.algocourses.ru/?task_id=5. Ограничения на входные данные: $1 \leq a, b, c \leq 1000$.

Гарантируется, что существует реализация функции без ветвлений (условного оператора, тернарного оператора и т.д.). Также Дед очень честный и не использовал в функции какие-то странные числа (например 1231, 8190, 52, ...).

Отправьте реализацию функцию `solve` в следующем формате:

```
int solve(int a, int b, int c) {  
    // YOUR CODE  
}
```

Ничего кроме функции `solve` сдавать **не нужно**.

Задача К. К рекордов

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

Полине подарили на новый год массив a длины n . Также, у неё есть счастливое число k .

Так как Полине очень дорог человек, который подарил ей этот массив, она хочет носить часть этого массива с собой. Поэтому Полина хочет выбрать такой подотрезок этого массива, который содержит ровно k **рекордов**. Разумеется, она не хочет носить слишком большой массив, поэтому Полина хочет выбрать самый короткий такой подотрезок, а среди самых коротких — самый правый.

Элемент массива называется **рекордом**, если он строго больше, чем все предыдущие элементы.

Формат входных данных

В первой строке задано два целых числа n и k — длина массива и счастливое число Полины ($1 \leq k \leq n \leq 10^5$).

В следующей строке находится n целых чисел — элементы массива ($1 \leq a_i \leq 10^9$).

Формат выходных данных

Если подотрезка с k **рекордами** в этом массиве не существует, выведите -1 .

Иначе выведите два числа l и r — номера элементов, с которых начинается и заканчивается минимальный по длине такой подотрезок. Если таких подотрезков несколько, необходимо вывести самый правый из них.

Примеры

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

Задача L. Украшение окон

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мегабайт

Перед новым годом преподавателю Ване поручили украсить окна. Он решил приклеить на окна n домиков разной высоты (высота i -го из них равна a_i). Преподаватель может уменьшать высоту домиков, но после каждого уменьшения на сантиметр ему становится чуть грустнее. Будем измерять печаль Вани в грустинках, тогда за одно уменьшение Иван получает одну грустинку.

Ваня выбрал порядок в каком расставить домики, чтобы получились красивые переходы цветов, однако он будет считать окна красивыми тогда и только тогда, когда будет ровно k домиков, которые строго выше соседних домов (если таковые есть). Например, если последовательность высот это 5, 4, 6, 2, то красота равна 2, так как выше своих соседей только домики с высотами 5 и 6.

Ваня хочет праздновать новый год в хорошем настроении, поэтому хочет получить как можно меньше грустинок. Однако, Ваня все еще не знает конкретное значение k . Вам предлагается посчитать нужную величину для всех возможных красот k в отрезке $1 \leq k \leq \lceil \frac{n}{2} \rceil$. Здесь $\lceil \frac{n}{2} \rceil$ обозначает n делённое на 2, округлённое вверх.

Формат входных данных

Первая строка ввода состоит из единственного целого n ($1 \leq n \leq 5000$)— количества домиков в последовательности.

Вторая строка содержит n целых чисел a_i ($1 \leq a_i \leq 100\,000$) — высоты домиков в последовательности.

Формат выходных данных

Выведите $\lceil \frac{n}{2} \rceil$ чисел, разделенных пробелами. i -ое выведенное число должно быть равно минимальному числу грустинок, которые получит Ваня, после изменения домиков, так чтобы красота украшения была равна i .

Примеры

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

Задача М. Время (TIME)

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

<https://timelksh.algocourses.ru/>

Задача N. Волшебный мешок

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

В далеком снежном королевстве жил великий волшебник — Дед Мороз. Каждый год он создавал особый Волшебный Мешок, чтобы доставить подарки детям по всему миру. В мешке были числа, каждое из которых символизировало силу новогоднего чуда. Чем больше это число, тем ярче светятся елки, громче смех детей и сильнее дух праздника.

Однако волшебство этих чисел подчинялось древнему закону числовой магии. Когда Дед Мороз складывал числа в своем мешке, их суммарная энергия должна была усиливаться на каждом этапе. Это измерялось величиной, которую древние математики называли наибольшим общим делителем (GCD). Чем больше этот GCD, тем мощнее становилась магия префиксных сумм — это значит, что каждое промежуточное сложение давало больше праздничной силы.

Но вот незадача: числа в мешке Деда Мороза оказались перепутаны! Чтобы подготовить подарки вовремя, Деду Морозу нужно найти такой порядок чисел, чтобы максимизировать магическую силу GCD их префиксных сумм.

Формат входных данных

В первой строке вводится целое число n — количество подарков ($1 \leq n \leq 10^5$).

Во второй строке вводятся n целых чисел — магические силы подарков ($1 \leq a_i \leq 10^9$).

Формат выходных данных

В первой строке выведите максимальный возможный GCD.

Во второй строке выведите перестановку длины n — оптимальный порядок подарков.

Примеры

стандартный ввод	стандартный вывод
3 2 6 4	2 2 1 3
4 5 1 6 9	1 4 3 2 1

Задача О. Хорошие раскраски New Year edition

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	10 секунд
Ограничение по памяти:	256 мегабайт

Эта задача даже не очень задача. От вас лишь требуется раскрасить картинки, которые мы вам даем в архиве. В нем есть некоторое количество примеров, как должны выглядеть раскрашенные и нераскрашенные изображения. Также в архиве лежат все тесты, на которых будут тестироваться ваши решения. Для вашего удобства в архиве также лежит файл «base.cpp», в котором показан пример того, как должен производиться ввод и вывод в этой задаче.

<http://ejudge.lksh.ru/olymp/paintings.zip>

Формат входных данных

Во входных данных вы получаете файл в формате ppm. Для вас уже был написан код, который вводит файл, сохраняет его в табличку и выводит его.

Формат выходных данных

В выходных данных выведете картинку в формате ppm.

Примеры

стандартный ввод	стандартный вывод
P3 654 736 255 ...	P3 654 736 255 ...
P3 677 717 255 ...	P3 677 717 255 ...

Замечание

В первом семпле отображается лишь начало файла «bird.ppm» так как он слишком большой.
Во втором семпле отображается лишь начало файла «totoro.ppm» так как он слишком большой.

Задача Р. Копим деньги

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

Дед мороз копит деньги на подарки всем к новому году, и поскольку он очень ленив деньги ему приносят только эльфы, живущие у него дома. К моменту нового года у деда мороза дома будут жить n эльфов, i -й эльф приходит в дом в момент времени T_i и приносит 1 монету каждые R_i секунды. За какое минимальное кол-во секунд дед мороз сможет накопить X монет для того, чтобы купить подарки.

Формат входных данных

Первая строка входных данных содержит два целых числа n, X ($1 \leq n \leq 10^5, 1 \leq X \leq 10^9$) — кол-во эльфов, сколько монет требуется.

Следующие n строк содержат два целых числа T_i, R_i ($1 \leq T_i \leq 10^9, 1 \leq R_i \leq 10^3$) — время когда эльф придет и его период.

Формат выходных данных

Выведите ответ на задачу))

Пример

стандартный ввод	стандартный вывод
2 13 5 4 3 2	21

Замечание

Ко времени 21 первый эльф принесет 4 монеты, второй эльф принесет 9 монеты.

Ко времени 11 первый эльф принесет 1 монету, второй эльф принесет 4 монеты, суммарно будет 5 монет, поэтому ответ не равен 11.

Задача S. Тайный Санта

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	4 секунды
Ограничение по памяти:	512 мегабайт

Шел 23956630-й год, и в очередную смену ЛКШ.Зима приехало аж n талантливых школьников. По традиции, культоргги решили провести Тайного Санту. Однако в этот раз (никогда не было, и вот опять!) они решили поэкспериментировать с форматом, поэтому придумали новые правила.

Перед сменой каждый школьник заготавливает один подарок, будем считать что i -й школьник готовит подарок с номером i . После этого, в Новый Год, культоргги несколько раз сделают следующее: они выберут двух школьников и попросят их обменяться имеющимися сейчас у них подарками.

Культоргги уже заготовили последовательность из m пар школьников, которые должны обменяться подарками, как вдруг поняли, что она им не нравится! Поэтому они решили оставить только некоторый подотрезок из этой последовательности. Как вы уже могли догадаться, у наших культоргов странные запросы.

Более конкретно, у них q запросов, задающихся тройками чисел l_i, r_i, x_i — узнать номер подарка, который в итоге получит школьник с номером x_i если просить обменяться подарками только пары с номерами от l_i до r_i . Поскольку у культоргов много других забот, они просят вас разобраться с этой задачей и ответить на все запросы.

Формат входных данных

В первой строке входных данных находятся числа n и m ($2 \leq n \leq 100$, $1 \leq m \leq 2 \cdot 10^5$) — количество школьников и количество пар школьников, которые заготовили культоргги. В следующих m строках перечислены пары a_i, b_i школьников, обменивающихся подарками ($1 \leq a_i, b_i \leq n$, $a_i \neq b_i$). В следующей строке находится одно число q — количество запросов ($1 \leq q \leq 10^5$). В каждой из следующих q строк находятся три числа l_i, r_i, x_i ($1 \leq l_i \leq r_i \leq m$, $1 \leq x_i \leq n$) — параметры очередного запроса.

Формат выходных данных

Выведите q чисел по одному на строке, i -е из которых должно означать номер подарка, который окажется в руках у x_i школьника, если последовательно попросить обменяться подарками все пары школьников с l_i до r_i .

Пример

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

Задача Т. Мандариновый чай

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

Вова привез с собой в смену ЛКШ.Зима несколько килограмм вкуснейшего новогоднего чая. И вот перед ним встала сложная задача: нужно решить какую долю чая он сейчас будет заваривать. Он хочет заваривать за раз строго меньше, чем $\frac{1}{q}$ -ю часть чая, иначе ему не хватит его запасов до конца смены, а Вова не хочет пить только кофе в новом году. С другой стороны, если он заварит не более, чем $\frac{1}{p}$ от всего чая, то напиток получится не достаточно крепким и преподаватель уснет прямо на паре. При этом всем, Владимир хочет разделить чай на не более чем n частей и каждый раз класть в заварочный чайник несколько из них. Вова хочет знать, сколько у него есть вариантов выбрать ту долю чая, которую он заварит сейчас.

Более формально, нужно посчитать количество таких дробей $\frac{a}{b}$ (необязательно несократимых), что $\frac{1}{p} < \frac{a}{b} < \frac{1}{q}$, причем a, b – целые числа и известно, что $b \leq n$.

Формат входных данных

В первой строке входных данных дано целое число m ($m \leq 10^5$) – количество тестов. В следующих m строках написано по три целых числа p_i, q_i, n_i ($1 \leq p_i, q_i, n_i \leq 10^9$)

Формат выходных данных

Для каждого теста вам нужно вывести по одному числу в строке, каждое из которых будет равно количеству таких дробей $\frac{a}{b}$ (возможно сократимых), что $\frac{1}{p_i} < \frac{a}{b} < \frac{1}{q_i}$ и $1 \leq b \leq n_i$

Пример

стандартный ввод	стандартный вывод
2	1
3 2 5	30
3 1 10	

Задача U.

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 5 секунд
Ограничение по памяти: 1024 мегабайта

algotup тулзы : скальп ,
бобан ,
фара , шогa , гибон , бланка
, выдра , каспер , валерия , пепел , вуда , барон
, рипер (не Джек)
, ёжик ,
постик
.