

Задача А. Фёдор и палиндромы

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Палиндромы, это такие строки, которые читаются в обоих направлениях одинаково. Например строка «abcba» является палиндромом.

Программист Фёдор настолько любит палиндромы, что решил использовать их для разработки собственной капчи, т.е. компьютерного теста, который помогает определить, кем является пользователь – человеком или компьютером.

Фёдор придумал следующий тест. Рассмотрим простой случай. Предлагается склеить две строки, и определить, можно ли из полученной строки построить палиндромом, если в ней при необходимости переставить несколько букв местами. Например, две строки «ab» и «cab» можно склеить в строку «abcab», которая после перестановки последних двух букв превращается в палиндром «abcba». Пользователю в интернете предлагается набор строк. Чтобы пройти проверку, ему нужно определить количество способов такой попарной склейки.

Но ведь такую проверку с лёгкостью может пройти и компьютер! Но Фёдор просто так не сдаётся и продолжает настаивать на надёжности своей проверки. Весомым аргументом для Фёдора будет являться программа, которая ответит правильно на все его тесты.

Формат входных данных

В первую строку входного файла записано одно целое число N – количество строк в тесте Фёдора ($2 \leq N \leq 10^5$).

Далее задаются N строк, целиком состоящих из маленьких букв латинского алфавита.

Гарантируется, что суммарная длина строк не превышает 10^6 символов.

Формат выходных данных

В выходной файл необходимо вывести одно целое число – количество пар строк, при соединении которых получаются строки, которые можно преобразовать в палиндром перестановкой некоторого числа букв.

Примеры

<code>input.txt</code>	<code>output.txt</code>
2 ab ba	1
5 a a ab c b	4

Задача В. Химические реакции

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

В параллельной реальности, как известно, действуют свои необычные законы физики. Некоторые законы химии зависят от законов физики, и поэтому они тоже могут быть на редкость необычными.

В одной такой реальности живет химик Ким Мих. У него есть несколько химических веществ. Он хочет знать, какие еще вещества можно получить, проводя реакции над теми веществами, которые у него уже есть. Ким весьма невнимателен, и может не учесть какое-нибудь полученное вещество. Кроме того, количество возможных реакций в реальности Кима может быть очень большим, поэтому ему легко запутаться.

Ким просит Вас помочь написать программу, способную найти вещества, которые он может получить из имеющихся у него веществ.

Известно, что химических веществ в реальности Кима лишь конечное число. Вещества могут взаимодействовать друг с другом: из двух различных веществ может получиться ровно одно новое. Гарантируется, что результат реакции не зависит от порядка реагентов, и не существует реакций с одинаковыми реагентами и разными результатами реакций.

Формат входных данных

В первой строке входного файла записаны через пробел два целых числа N и M , где N – это количество веществ в реальности Кима, M – количество веществ, которые есть у Ким Муха ($1 \leq M \leq N \leq 10^5$).

Во второй строке дано единственное целое число K – количество возможных реакций ($1 \leq K \leq 10^5$).

В следующих K строках описаны эти реакции. Каждая реакция задается в виде трех целых чисел, записанных через пробел: a, b, c – при взаимодействии веществ с номерами a и b получится вещество с номером c ($1 \leq a, b, c \leq N$).

В последней строке через пробел даны M целых чисел v_1, \dots, v_M – номера веществ, которые есть у Кима Муха изначально.

Формат выходных данных

В первую строку выходного файла необходимо вывести целое число T – количество веществ, которые может получить Ким из имеющихся у него. Это число должно включать и количество веществ, которые были у Кима изначально.

Во вторую строку нужно вывести через пробел T чисел – номера веществ, которые он сможет получить.

Примеры

<code>input.txt</code>	<code>output.txt</code>
5 2 3 1 2 3 2 3 4 3 4 5 1 4	2 1 4
5 2 3 1 2 3 2 3 4 3 4 5 1 2	5 1 2 3 4 5

Задача С. Весёлые сообщения

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Учёные из института компьютерной лингвистики занимаются анализом сообщений в социальных сетях. В частности, они ищут зависимость количества лайков от содержания сообщения. Недавно ученые выяснили, что наибольшее количество лайков получают самые весёлые сообщения. Чтобы использовать научный подход при изучении текстов и измерять, насколько данное сообщение весёлое, они ввели параметр «весёлость» строки. Этот параметр вычисляется следующим образом:

- если в строке нет символов ')', то весёлость этой строки равна 0;
- если в строке есть символы ')', то для каждой последовательности подряд идущих символов ')' нужно вычислить квадрат длины этой последовательности, после чего сложить полученные числа.

Например, весёлость строки «dobroe))utro)))» равна $2^2 + 3^2 = 13$, поскольку в ней есть две последовательности подряд идущих символов ')' длины 2 и 3, соответственно.

Учёные хотят использовать свои результаты, чтобы в группе их института в социальной сети было очень много лайков. Они просят Вас написать программу, которая бы преобразовывала сообщения в максимально весёлые. Для этого программа может удалить некоторое количество символов. Но, так как сообщения не должны потерять своё основное содержание, то количество удаляемых символов ограничено, и не может быть больше заданного числа.

Формат входных данных

В первой строке входного файла записаны два целых числа N и K – количество символов в исходном сообщении и количество символов, которое может удалить программа ($1 \leq N \leq 10^5$, $0 \leq K \leq 200$).

Во второй строке задается сообщение, которое может содержать цифры, буквы латинского алфавита и следующие знаки препинания: '.', ',', ')', '(', '!', '?'. Известно, что количество символов ')' в исходном сообщении не превышает 200.

Формат выходных данных

В выходной файл необходимо вывести одно целое число – значение максимальной весёлости исходного сообщения, которого можно достичь с помощью удаления из него не более, чем K символов.

Примеры

input.txt	output.txt
37 9 privet!)))))kakdela?)ahaha)))))poKa	97
33 4 ura!))final))olimpiada)))))eee	54

Задача D. Антон и xor

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Антон только-только начал заниматься программированием, но уже достиг определённых успехов. Ещё вчера он умел выполнять лишь тривиальные арифметические операции и писать простенькие циклы, а сегодня уже научился вычислять операцию *xor* для двух целых чисел.

Конечно, как любой прилежный ученик, Антон хочет закрепить полученные знания на примере конкретной задачи. Со вчерашних занятий у него остался набор целых чисел. Антон решил модифицировать этот набор следующим образом: он хотел бы выбрать некоторое целое неотрицательное число x , и выполнить операцию $a_{new} := a \text{ xor } x$ для всех элементов a из вчерашнего набора. Так как Антон планирует использовать новый набор в дальнейших разработках, он бы хотел, чтобы при работе с ним не возникало проблем с переполнением чисел. Для этого он хочет выбрать такое минимальное число x , чтобы сумма элементов новой последовательности была минимальна.

Поскольку Антон только начал делать свои первые шаги в мире программирования, то попросил Вас написать программу, которая бы находила такое число x .

Формат входных данных

В первой строке входного файла записано единственное целое число N – количество чисел в наборе Антона ($1 \leq N \leq 10^5$).

Следующие N строк описывают исходный набор чисел Антона. Каждая строка содержит ровно два целых числа M_i и K_i , означающие, что число M_i встречается в изначальном наборе K_i раз ($1 \leq i \leq N$, $0 \leq M_i \leq 10^{18}$, $1 \leq K_i \leq 10^{12}$).

Формат выходных данных

В выходной файл требуется вывести одно целое неотрицательное число x такое, что сумма элементов нового набора, построенного с помощью x и операции *xor*, будет минимальна. Если вариантов ответа несколько, необходимо выбрать минимальное x .

Примеры

<code>input.txt</code>	<code>output.txt</code>
5 0 5 4 1 3 1 2 1 1 1	0
3 2 5 1 3 3 9	3

Замечание

Операция *xor* (побитовое исключающее ИЛИ) – это такая операция над двумя целыми числами, при которой i -й разряд результата в двоичной системе счисления будет равен 1 тогда и только тогда, когда ровно у одного из двух исходных целых чисел в i -м разряде в двоичной системе счисления стоит 1.

Задача Е. Прямоугольный треугольник

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Однажды, возвращаясь из школы после урока геометрии, Гриша увидел на рекламных вывесках торгового центра три целых числа. Продолжая свой путь домой, он немного изменил в уме одно из чисел и получил три числа, являющиеся длинами сторон некоторого прямоугольного треугольника. Далее он замечал ещё несколько раз новые числа, каждый раз ему удавалось, проделав те же манипуляции, получить длины сторон прямоугольного треугольника.

В итоге у него возникла гипотеза о том, что существует некоторое минимальное число, на которое можно изменить одно из данных трех чисел, чтобы получились длины сторон прямоугольного треугольника. Но Гриша до сих пор помнит, как его английский друг Эндрю, поторопившись, совершил ошибку в доказательстве очень простой теоремы.

И пока он пишет текст будущей прорывной статьи, Вам предлагается написать программу, которая проверяет его гипотезу в следующем виде. На вход программе подается четыре числа a , b , c и k . Программа должна определить, можно ли изменить одно из трех чисел a , b или c не более чем на k , и получить длины сторон прямоугольного треугольника.

Формат входных данных

В первой строке входного файла задано одно целое число T – количество тестовых наборов ($1 \leq T \leq 1000$).

В каждой из следующих T строк записано по одному тестовому набору, содержащему четыре целых числа a , b , c и k , для которых нужно проверить гипотезу Гриши ($1 \leq a, b, c, k \leq 10^9, \min(a, b, c) > k$).

Формат выходных данных

В выходной файл необходимо вывести для каждого тестового набора в отдельную строку слово "YES", если можно изменить одно из трех чисел a , b или c не более чем на k , и получить длины сторон прямоугольного треугольника, либо слово "NO", в противном случае.

Примеры

input.txt	output.txt
2	YES
3 3 5 1	YES
2 2 2 1	
1	NO
100 100 100 1	