

Задача А. Словарь

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1.5 секунд
Ограничение по памяти: 256 мегабайт

Дан набор слов и текст, требуется определить для каждого слова, присутствует ли оно в тексте как подстрока.

Формат входных данных

В первой строке дан текст (не более 10^6 строчных латинских букв). Далее дано число M — количество слов в словаре.

В следующих M строках записаны слова (каждое длиной не более 30 строчных латинских букв). Слова различны и отсортированы в лексикографическом порядке.

Суммарная длина слов в словаре не более 10^5 .

Формат выходных данных

M строк вида Yes, если слово присутствует, и No иначе.

Примеры

стандартный ввод	стандартный вывод
trololo 3 abacabadabacaba olo trol	No Yes Yes
abacabadabacabaabacabadabacaba 1 abacabadabacabaabacabadabacaba	Yes

Задача В. Библиотека

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

Прошел почти год с момента, как Рик оказался на Флорине, однако его сознание никак не прояснялось. Воспоминания о прошлом были спрятаны в глубинах его разума, а может и вовсе утеряны. Однако сегодня что-то случилось. Рик вспомнил: у него была работа. Он анализировал Ничто. Наверное, Ничто — это космос, а значит Рик в прошлом был космоаналитиком. А еще Рик вспомнил, что все жители Флорины должны были погибнуть, но он не знал, почему.

Резидента Мирлина Теренса заинтересовала эта информация, поэтому он взял Рика с собой в библиотеку Верхнего города. Может быть, какая-нибудь литература по космоанализу могла бы вернуть ему память? Теренс не знал, что пропавшего космоаналитика активно ищут, а потому в библиотеке был получен приказ сообщать о любых посетителях, которые спросят о такой литературе. Библиотекарь отследил запросы наших героев в поисковой системе и поспешил вызвать патрульных.

Тем временем Теренс предложил Рiku ознакомиться с книгой известного автора Врийта "Трактат об инструментальном космоанализе". Рiku книга определенно показалась знакомой, особенно его привлекла строка s . Смысла самой строки, он, к сожалению, не понимал, однако в ее частях он видел что-то знакомое. Чтобы разобраться подробнее, Рик решил изучить все подстроки s . Однако изучать равные подстроки не было смысла, а остальные стоило как-либо систематизировать. Например, расставить их по длине и в алфавитном порядке. Поэтому Рик попросил вас узнать, сколько у данной строки существует пар подстрок s_1 и s_2 равной длины, таких, что $s_1 < s_2$ лексикографически.

Формат входных данных

Задана строка s , состоящая из строчных латинских букв ($|s| \leq 2500$).

Формат выходных данных

Выведите одно число — количество искомых пар подстрок.

Пример

стандартный ввод	стандартный вывод
abac	9

Замечание

Рассмотрим подстроки длины 1. Имеется две подстроки "a", каждая из которых меньше подстрок "b" и "c". Также подстрока "b" меньше подстроки "c". Отсюда получаем 5 пар искомых подстрок.

Теперь рассмотрим подстроки длины 2. Подстрока "ab" меньше подстрок "ba" и "ac", а строка "ac" меньше, чем строка "ba". Отсюда получаем еще 3 пары.

Наконец, рассмотрим подстроки длины 3. Подстрока "aba" меньше подстроки "bac".

Таким образом, суммарно получаем 9 искомых пар подстрок.

Задача С. К-я строка

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

Реализуйте структуру данных, которая поддерживает следующие операции:

- добавить в словарь строку S ;
- найти в словаре k -ю строку в лексикографическом порядке.

Изначально словарь пуст.

Формат входных данных

Первая строка входного файла содержит число N — количество команд ($1 \leq N \leq 10^5$).

Последующие N строк содержат по одной команде каждая. Команды записываются следующим образом:

- 1 S — добавить строку S в словарь;
- 2 k — вывести k -ю строку в лексикографическом порядке.

Гарантируется, что при запросе k -й строки она существует. Также гарантируется, что сумма длин всех добавляемых строк не превышает $3 \cdot 10^5$.

Все строки состоят из строчных латинских букв.

Формат выходных данных

Для каждого запроса второго типа выведите k -ю в лексикографическом порядке строчку из словаря на момент запроса.

Гарантируется, что суммарная длина строк в выходном файле не превышает 10^5 .

Пример

стандартный ввод	стандартный вывод
7	tolstoy
1 pushkin	gogol
1 lermontov	
1 tolstoy	
1 gogol	
1 gorkiy	
2 5	
2 1	

Задача D. Мультимножество Василия

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

У автора уже закончились истории про Василия, поэтому он просто написал формальную постановку задачи.

У вас есть q запросов и мультимножество A , изначально содержащее только число 0. Запросы бывают трёх видов:

- «+ x » — добавить в мультимножество A число x .
- «- x » — удалить одно вхождение числа x из мультимножества A . Гарантируется, что хотя бы одно число x в этот момент присутствует в мультимножестве.
- «? x » — вам даётся число x , требуется вычислить $\max_{y \in A} x \oplus y$, то есть максимальное значение побитового исключающего ИЛИ (также известно как XOR) числа x и какого-нибудь числа y из мультимножества A .

Мультимножество — это множество, в котором разрешается несколько одинаковых элементов.

Формат входных данных

В первой строке входных данных содержится число q ($1 \leq q \leq 200\,000$) — количество запросов, которые требуется обработать Василию.

Каждая из последующих q строк входных данных содержит один трёх символов «+», «-» или «?» и число x_i ($1 \leq x_i \leq 10^9$). Гарантируется, что во входных данных встречается хотя бы один запрос «?».

Обратите внимание, что число 0 всегда будет присутствовать в мультимножестве.

Формат выходных данных

На каждый запрос типа «?» выведите единственное целое число — максимальное значение побитового исключающего ИЛИ для числа x_i и какого-либо числа из мультимножества A .

Пример

стандартный ввод	стандартный вывод
10	11
+ 8	10
+ 9	14
+ 11	13
+ 6	
+ 1	
? 3	
- 8	
? 3	
? 8	
? 11	

Замечание

После первых пяти операций в мультимножестве A содержатся числа 0, 8, 9, 11, 6 и 1.

Ответом на шестой запрос будет число $11 = 3 \oplus 8$ максимальное из чисел $3 \oplus 0 = 3$, $3 \oplus 9 = 10$, $3 \oplus 11 = 8$, $3 \oplus 6 = 5$ и $3 \oplus 1 = 2$.

Задача Е. Странный сон

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

Однажды Константин, поучаствовав в очередной, уже 13-й по счету международной олимпиаде, возвращался на поезде домой. Он как всегда сидел и размышлял о смысле жизни, попутно решая задачи по программированию. Через некоторое время Константин задремал, но вот беда, для того, чтобы проснуться, он должен решить всплывшую у него в голове задачу, не дающую ему покоя!

В этот раз Константину приснилось дерево, изначально состоящее всего из одной вершины с номером 1. В поставленной им задаче к дереву постепенно добавлялись новые вершины. В i -ю секунду в дерево добавлялась вершина с номером $i + 1$, которая подвешивалась в качестве сына к вершине p_i , а на ребре между вершинами $i + 1$ и p_i записывалась буква c_i .

Каждому пути из корня дерева до вершины v соответствует некоторая строка, получающаяся путем выписывания символов, записанных на ребрах текущего пути в порядке следования от корня к вершине v . Перед Константином стояла нелегкая на первый взгляд задача — после каждого добавления новой вершины посчитать количество уникальных строк, начинающихся в корне дерева (вершине с номером 1), и заканчивающихся в какой-либо другой вершине.

В своем сне Константин вовсе не гений, поэтому решить эту задачу сам он не в силах. Помогите Константину решить задачу и тем самым проснуться.

Формат входных данных

В первой строке записано число n — количество запросов на добавление новой вершины в дерево ($1 \leq n \leq 3 \cdot 10^5$).

В следующих n строках описаны запросы добавления вершин. i -й запрос описывается параметрами p_i ($1 \leq p_i \leq i$) и c_i , которые означают, что добавленная вершина с номером $i + 1$ подвешивается к вершине с номером p_i в качестве потомка, а на полученном ребре записывается символ c_i — строчная буква латинского алфавита.

Формат выходных данных

Выведите n строк. В i -й строке выведите ответ на задачу Константина после добавления $(i + 1)$ -й вершины.

Примеры

стандартный ввод	стандартный вывод
2 1 b 2 p	1 2
3 1 o 1 o 2 j	1 1 2

Задача F. N-функция

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

Вам дана строка из строчных латинских букв. От вас требуется вычислить N-функцию данной строки.

Формат входных данных

Вводится строка, состоящая из строчных латинских букв. Длина строки не превышает 10^6 .

Формат выходных данных

Требуется вывести N-функцию данной строки.

Пример

стандартный ввод	стандартный вывод
abacaba	7 0 1 0 3 0 1

Замечание

Предполагается, что значение N-функции для первого символа равно длине строки.

Задача G. Префикс-функция

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

Вам дана строка из строчных латинских букв. От вас требуется вычислить префикс-функцию данной строки.

Формат входных данных

Вводится строка, состоящая из строчных латинских букв. Длина строки не превышает 10^6 .

Формат выходных данных

Требуется вывести префикс-функцию данной строки.

Пример

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

Замечание

Предполагается, что значение префикс-функции для первого символа равно нулю.

Задача Н. Поиск подстроки (1 балл)

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

Найти все вхождения строки T в строку S .

Формат входных данных

Первые две строки входных данных содержат строки S и T , соответственно. Длины строк больше 0 и меньше 500 000, строки содержат только строчные латинские буквы.

Формат выходных данных

Выведите номера символов, начиная с которых строка T входит в строку S , в порядке возрастания.

Пример

стандартный ввод	стандартный вывод
ababbababa aba	0 5 7

Задача I. Строчечки

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

Мальчик Кирилл написал однажды на листе бумаги строчку, состоящую из больших и маленьких латинских букв, а после этого ушел играть в футбол. Когда он вернулся, то обнаружил, что его друг Дима написал под его строкой еще одну строчку такой же длины. Дима утверждает, что свою строчку он получил циклическим сдвигом строки Кирилла на несколько шагов вправо (циклический сдвиг строки «abcde» на 2 позиции вправо даст строку «deabc»). Однако Дима известен тем, что может случайно ошибиться в большом количестве вычислений, поэтому Кирилл в растерянности – верить ли Диме? Помогите ему! По данным строкам выведите минимальный возможный размер сдвига или -1 , если Дима ошибся.

Формат входных данных

Первые две строки входных данных содержат строки Кирилла и Димы, соответственно. Длины строк одинаковы, не превышают 10^6 и не равны 0.

Формат выходных данных

Выведите единственное число – ответ на вопрос задачи.

Пример

стандартный ввод	стандартный вывод
zabcd abcdz	4

Задача J. А-функция от строки

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

Дана строка S , состоящая из N символов. Определим функцию $A(i)$ от первых i символов этой строки следующим образом:

$A(i) :=$ максимально возможному k , что равны следующие строки:

$S_1 + S_2 + \dots + S_k$ и $S_i + S_{i-1} + \dots + S_{i-k+1}$, где S_i — i -ый символ строки S , а знак $+$ означает, что символы записываются в строчку непосредственно друг за другом.

Напишите программу, которая вычислит значения функции A для заданной строки для всех возможных значений i от 1 до N .

Формат входных данных

В первой строке входного файла записано одно число N . $1 \leq N \leq 200\,000$. Во второй строке записана строка длиной N символов, состоящая только из больших и/или маленьких латинских букв.

Формат выходных данных

В выходной файл выведите N чисел — значения функции $A(1), A(2), \dots, A(N)$.

Пример

стандартный ввод	стандартный вывод
5 aabaa	1 2 0 1 5

Задача К. Поиск подмассива со сдвигом

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1.5 секунд
Ограничение по памяти: 256 мегабайт

Даны два массива a и b , состоящие из натуральных чисел. Требуется найти все вхождения второго массива в первый с точностью до сдвига на какое-то целое число. Формально, корректным вхождением является индекс l такой, что для каждого индекса $i \in [l, l + k)$ $a[l + i] = b[i] + d$, где d - целое число, фиксированное для всех i .

Формат входных данных

В первой строке входного файла даны два числа n, k ($1 \leq n, k \leq 10^5$) — длины первого и второго массивов соответственно. Во второй строке входного файла даны n чисел a_i через пробел, а в третьей строке - k чисел b_i , описывающие элементы массивов ($0 \leq a_i, b_i \leq 10^9$).

Формат выходных данных

В единственной строке выходного файла выведите все индексы вхождений в возрастающем порядке. Числа требуется выводить через пробел.

Примеры

стандартный ввод	стандартный вывод
5 2 1 2 4 5 7 10 11	1 3
10 3 1 2 1 2 1 10 10 15 16 15 1 2 1	1 3 8

Задача L. Глупые Песни

Имя входного файла: *стандартный ввод*
Имя выходного файла: *стандартный вывод*
Ограничение по времени: 4 секунды
Ограничение по памяти: 256 мегабайт

KiKoS является фанатом телесериала «Как я встретил вашу маму». Недавно он обнаружил плейлист Spotify с его саундтреком.

Помимо неоспоримых хитов «Давайте пойдем в торговый центр» и «Ничто не подходит мне так, как костюм», он нашел очень глупую песню: «Bang bang bangity bang». Это, по сути, единственная фраза в всей песне. Чтобы сделать ее длиннее, нужно просто повторить ту же строку, но добавив «Я сказал» в начале, и тогда можно снова спеть все это.

Как новичок-песенник, KiKoS хочет использовать эту структуру для новой песни. Формально, дано непустое слово S , состоящее из строчных английских букв, песня S состоит из **блока из двух строк**, который можно повторять несколько раз, используя то же слово S . Каждый блок выглядит так:

S , S Sity S
i said S , S Sity S

Дано большое многострочное текстовое сообщение, определите максимальное возможное количество символов в некоторой последовательности последовательных блоков, которые образуют действительную песню.

Формат входных данных

Входные данные состоят из нескольких строк, состоящих из строчных английских букв, запятых и пробелов. Каждая строка заканчивается символом новой строки. Общее количество символов, включая символы конца строк, не превышает 10^5 . Пустых строк нет, и строки не начинаются и не заканчиваются пробелом.

Формат выходных данных

Выведите одно целое число: максимальное возможное количество символов в некоторой последовательной последовательности блоков, которые соответствуют шаблону песни. Это включает символы новой строки в конце каждой строки последовательности, включая последнюю строку песни. Каждый символ новой строки считается 1 символом: обратите внимание на это, если у вас Windows.

Если нет блоков, соответствующих шаблону песни, выведите -1 .

Примеры

<i>стандартный ввод</i>	<i>стандартный вывод</i>
bang, bang bangity bang i said bang, bang bangity bang	55
bang, bang bangity bang i said bang, bang bangity bang bank, bank bankity bank i said bank, bank bankity bank bank, bank bankity bank i said bank, bank bankity bank	110
grandmasters are red, experts are blue, parsing problems are bad, but we believe in you	-1