

## Задача А. Наскальная живопись

Имя входного файла:            стандартный ввод  
Имя выходного файла:        стандартный вывод  
Ограничение по времени:    0.3 секунд  
Ограничение по памяти:      128 мегабайт

Доктор Брэйвстоун вместе с компанией нашел в Джуманджи древнюю пещеру. На стене этой пещеры была написана строка  $s$ , состоящая из строчных латинских букв. Теперь, чтобы продолжить прохождение, исследователям нужно определить максимальную длину палиндрома, который является подстрокой некоторого циклического сдвига строки  $s$ .

Палиндромом называется строка, которая читается одинаково как слева направо, так и справа налево. Например, строки *abacaba*, *abba* и *q* — являются палиндромами. Подстрокой строки называется некоторая последовательность подряд идущих символов этой строки. Циклическим сдвигом строки называется строка, полученная из исходной отрезанием некоторого префикса и дописыванием его в конец строки. Например, строки *bacabaa*, *sabaaba* и *abacaba* являются циклическими сдвигами строки *abacaba*.

### Формат входных данных

В единственной строке дана непустая строка  $s$ , состоящая из строчных латинских букв, длина которой не превышает  $10^6$ .

### Формат выходных данных

Выведите единственное число — искомую максимального длину палиндрома.

### Пример

стандартный ввод	стандартный вывод
babaca	5

## Задача В. Анаграммные подстроки

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

У Андриюши есть маленькая сестричка Аня. Она любит писать сообщения своему другу Гоше. Она хочет, чтобы никто не мог прочитать ее сообщения, поэтому она шифрует их подстановочным шифром. Подстановочный шифр заменяет каждый символ в сообщении на какой-либо еще, при этом равные символы заменяются на равные, а различные — на различные. Например, при шифровании с помощью подстановочного шифра  $e - a, l - b, o - w, v - c$  слово «love» оказывается зашифровано как «bwca».

Андриюша недавно перехватил одно из Аниных сообщений  $t$  и хочет выяснить, встречается ли там текст  $s$ . А именно, он хочет найти все позиции  $i$ , такие что существует подстановочный шифр, такой что  $t_i \dots t_{i+|s|-1}$  представляет собой зашифрованную версию  $s$ . Будем называть такие позиции потенциальными вхождениями  $s$  в  $t$ . Помогите Андриюше найти все потенциальные вхождения  $s$  в  $t$ .

### Формат входных данных

Первая строка входного файла содержит  $t$ . Вторая строка входного файла содержит  $s$ . Каждая строка состоит из символов с ASCII кодами от 33 до 126. Длина  $s$  не превышает длины  $t$ . Длина  $t$  не превышает 200 000. Обе строки непусты.

### Формат выходных данных

Первая строка выходного файла должна содержать  $k$  — количество потенциальных вхождений  $s$  в  $t$ . Вторая строка должна содержать  $k$  целых чисел — позиции потенциальных вхождений. Позиции в строке нумеруются, начиная с 1. Позиции следует перечислить в возрастающем порядке.

### Примеры

стандартный ввод	стандартный вывод
abacabadabacaba aba	7 1 3 5 7 9 11 13
abacabadabacaba love	0

## Задача С. Подстрока

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

Недавно разведка перехватила зашифрованное сообщение — строку  $s$ . Все ресурсы аналитического центра, в котором вы работаете, были брошены на его декодирование. Ваш отдел занимается шифрами нового поколения. На данный момент известно всего  $n$  таких шифров. Для каждого из них есть три характерных параметра — целые числа  $l, r$  и строка  $t$ . Пусть строка  $g$  была получена в результате применения этого метода. Тогда строка  $g_l g_{l+1} \dots g_{r-1} g_r$  (здесь  $g_i$  — это  $i$ -й символ строки  $g$ ) содержит  $t$  как подстроку.

Вам поручено определить для каждого типа шифрования, могло ли сообщение  $s$  быть получено в результате его применения.

### Формат входных данных

Первая строка входного файла содержит строку  $s$  ( $1 \leq |s| \leq 100\,000$ , где  $|s|$  — длина строки  $s$ ).

Вторая строка входного файла содержит целое число  $n$  — количество типов шифрования ( $1 \leq n \leq 100\,000$ ). Последующие  $n$  строк содержат по два целых числа  $l_i, r_i$  и строку  $t_i$ , разделенные пробелами — характерные параметры  $i$ -го метода шифрования ( $1 \leq l_i \leq r_i \leq |s|$ ).

Все строки состоят из строчных букв латинского алфавита. Суммарная длина всех  $t_i$  не превосходит 100 000.

### Формат выходных данных

Выведите одну строку — для каждого типа шифрования «+», если сообщение  $s$  могло быть получено в результате его применения, или «-» в противном случае.

### Пример

стандартный ввод	стандартный вывод
frommarsiam 3 6 10 i 2 11 am 1 9 human	++-

## Задача D. Лотерея

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

На одном из телеканалов каждую неделю проводится следующая лотерея. В течение недели участники делают свои ставки. Каждая ставка заключается в назывании какого-либо  $M$ -значного числа в системе счисления с основанием  $K$  (то есть, по сути, каждый участник называет  $M$  цифр, каждая из которых лежит в диапазоне от 0 до  $K - 1$ ). Ведущие нули в числах допускаются.

В некоторый момент прием ставок на текущий розыгрыш завершается, и после этого ведущий в телеэфире называет выигравшее число (это также  $M$ -значное число в  $K$ -ичной системе счисления). После этого те телезрители, у кого первая цифра их числа совпала с первой цифрой числа, названного ведущим, получают выигрыш в размере  $A_1$  рублей. Те, у кого совпали первые две цифры числа — получают  $A_2$  рублей (при этом если у игрока совпала вторая цифра, но не совпала первая, он не получает ничего). Аналогично угадавшие первые три цифры получают  $A_3$  рублей. И так далее. Угадавшие все число полностью получают  $A_M$  рублей. При этом если игрок угадал  $t$  первых цифр, то он получает  $A_t$  рублей, но не получает призы за угадывание  $t - 1, t - 2$  и т.д. цифр. Если игрок не угадал первую цифру, он не получает ничего.

Напишите программу, которая по известным ставкам, сделанным телезрителями, находит число, которое должна назвать телеведущая, чтобы фирма-организатор розыгрыша выплатила в качестве выигрышей минимальную сумму. Для вашего удобства ставки, сделанные игроками, уже упорядочены по неубыванию.

### Формат входных данных

В первой строке задаются числа  $N$  (количество телезрителей, сделавших свои ставки,  $1 \leq N \leq 100\,000$ ),  $M$  (длина чисел,  $1 \leq M \leq 10$ )  $K$  (основание системы счисления  $2 \leq K \leq 10$ ). В следующей строке записаны  $M$  чисел  $A_1, A_2, \dots, A_M$ , задающих выигрыши в случае совпадения только первой, первых двух, ..., всех цифр ( $1 \leq A_1 \leq A_2 \leq \dots \leq A_M \leq 100\,000$ ). В каждой из следующих  $N$  строк записано по одному  $M$ -значному  $K$ -ичному числу. Числа идут в порядке неубывания.

### Формат выходных данных

В первой строке выведите искомое число (если решений несколько — выведите минимальное), а во второй строке — сумму, которую при назывании телеведущей первого числа придется выплатить в качестве выигрыша.

### Примеры

стандартный ввод	стандартный вывод
10 3 2 1 3 100 000 000 001 010 100 100 100 100 110 111	011 6
1 1 10 100 0	1 0

## Задача E. Задачка на строчечки

Имя входного файла: inputik.txt  
Имя выходного файла: outputik.txt  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайтов

В стандартном поточике вводика или файлик  
е  
кой. На следующей строчечке программч  
ь а i  
н к; эти словечки мы назовём словарик п  
е о о н р  
н в utputik.txt N строчечек. В i-ой м а u  
д о о . й t  
о к - количюсик (сколько штукеч с д i  
х у к е т В ё k  
с б и о ем через пробельчик для к р а т .  
и л к т ) о ш t  
х й л а в всех вхожденъечек к ч а ч х  
м и а е з о а в е и t  
ё к ф с к ртированном про э ж х ч п с  
в с и , и о я т д о к р е в  
о н в ч й л с рочечек нач д о о ж е о л а  
з и о а т т и о й г д г к ш  
а т и е к ч о с ничек. н ч о е п р о а  
н а л о ъ а и а к с н р а  
л и в н н в х д е с я стю е т в ъ о м N п  
ы р е е . р х е г м , р  
м х а е н и ю св и к и с к е д н и о о ч р о o  
и к п д к у ч ж e a ч a г  
ю к и о и к ъ н е н д o x c и в и к ч e д k m k p  
у ъ д : x c e м a v a  
р н o k c к e d n и e т и д e в ы в а к ч и ъ н c o m  
о e в o и т ч д c m  
т л ы л в а к и р a в o л c з и i и k ч e ч o p k o л o  
о a в e a л e ч  
к м с и ч o к ъ л o x c e n и т c e в ы в a н ж л o d ж д k  
к н у a  
, e e ч и ч o т o п ъ и н т p a d n a t c a n и т c e в ы в a ю  
к ж щ н  
о х e т з и у к ч e в o л c o п - х a k ч e ч o p t c N х и a  
в й  
о к у б х и k c н и т a л х и к ъ н e л a m з и u k ч e ч o p t c т e d

## Формат входных данных

В стандартном поточике вводика или файлике `inputik.txt` ваша программочка найдёт строчечку из маленьких латинских буковок, которую мы назовём исходненькой. На следующей строчечке программочка найдёт числище  $N$  ( $1 \leq N \leq 1\,000\,000$ ), а в следующих  $N$  строчечках — по словечку из тех же маленьких латинских буковок; эти словечки мы назовём словариком. Суммарненькая суммочка длинниц словечек из словарика не превосходит  $1\,000\,000$ .

## Формат выходных данных

Ваша программочка должна вывести на стандартный поточичек выводика или в файл `outputik.txt`  $N$  строчечек. В  $i$ -ой строчечке программочка должна вывести несколько чиселок: первое чиселко — количюсик (сколько штучечек) вхожденьчек строчечки  $i$  из словарика в исходненькой, затем через пробельчик для каждого вхожденьчика выведите индексики началиков всех вхожденьчик этой строчечки в исходненькую в отсортированном порядочке. Индексики всех строчечек начинаются с единичек. Няшечки-преподавашечки гарантируют, что колчюсик вхожденьчик не превосходит  $1\,000\,000$ .

## Пример

<code>inputik.txt</code>	<code>outputik.txt</code>
<code>abrachkacadabrachka</code>	<code>2 1 12</code>
<code>4</code>	<code>1 9</code>
<code>abrachka</code>	<code>2 1 12</code>
<code>cadabrachka</code>	<code>0</code>
<code>ab</code>	
<code>marazmik</code>	

## Задача F. Словарь

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 1024 мегабайта

*Словарь* — это множество слов. Вы должны уметь обрабатывать запросы трех типов:

- «+ word» — добавить слово **word** в словарь, если оно в нем не присутствует.
- «- word» — удалить слово **word** из словаря, если оно там присутствует.
- «? text» — вычислить суммарное количество вхождений всех слов из словаря в текст **text**, при этом, если слово входит в текст несколько раз, то необходимо учесть каждое вхождение.

Гарантируется, что любое слово или текст являются непустыми строками, состоящими из букв **a**, **b** и **c**, суммарная длина которых не превосходит  $L$ . Однако, для упрощения задачи перед выполнением каждого запроса необходимо поступить следующим образом: пусть  $x$  обозначает ответ на последний запрос **?**, или 0, если таких запросов еще не было. Тогда необходимо очередную строку (**word** или **text**) циклически сдвинуть  $x$  раз. Напомним, что циклическим сдвигом строки  $s = s_0s_1 \dots s_{|s|}$  называется строка  $s' = s_1 \dots s_{|s|}s_0$ .

### Формат входных данных

В первой строке дано одно число  $Q$  — число запросов. В следующих  $Q$  строках находятся запросы. Суммарная длина строк во всех запросах не превосходит  $L$  ( $L \leq 5\,000\,000$ )

### Формат выходных данных

Для каждого запроса «?» выведите одно число — ответ на него.

### Пример

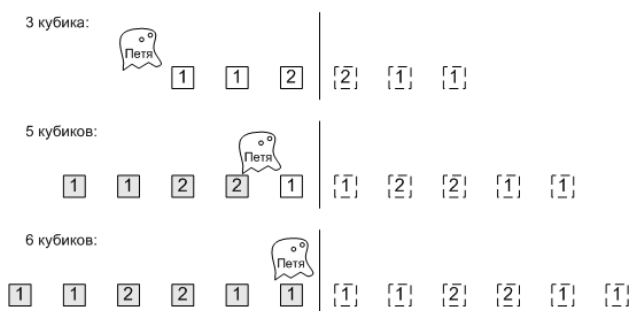
стандартный ввод	стандартный вывод
11	0
+ a	6
+ a	5
- a	7
- ab	
? abca	
+ ab	
+ a	
? abaaabb	
? aaabbab	
+ baa	
? babaca	

## Задача G. Кубики

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

Привидение Петя любит играть со своими кубиками. Он любит выкладывать их в ряд и разглядывать своё творение. Однако недавно друзья решили подшутить над Петей и поставили в его игровой комнате зеркало. Ведь всем известно, что привидения не отражаются в зеркале! А кубики отражаются.

Теперь Петя видит перед собой  $N$  цветных кубиков, но не знает, какие из этих кубиков настоящие, а какие — всего лишь отражение в зеркале.



Помогите Пете! Выясните, сколько у него может быть кубиков. Петя видит отражение всех кубиков в зеркале и часть кубиков, которая находится перед ним. Часть кубиков может быть позади Пети, их он не видит.

### Формат входных данных

Первая строка входного файла содержит два целых числа:  $N$  ( $1 \leq N \leq 100\,000$ ) и количество различных цветов, в которые могут быть раскрашены кубики, —  $M$  ( $1 \leq M \leq 100\,000$ ). Следующая строка содержит  $N$  целых чисел от 1 до  $M$  — цвета кубиков.

### Формат выходных данных

В выходной файл выведите в порядке возрастания все такие  $K$ , что у Пети может быть  $K$  кубиков.

### Пример

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



## Задача Н. Неточное совпадение

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

Даны строки  $p$  и  $t$ . Требуется найти все вхождения строки  $p$  в строку  $t$  в качестве подстроки с точностью до возможного несовпадения одного символа.

### Формат входных данных

Первая строка входного файла содержит  $p$ , вторая —  $t$  ( $1 \leq |p|, |t| \leq 10^6$ ). Строки состоят из букв латинского алфавита.

### Формат выходных данных

В первой строке выведите количество вхождений строки  $p$  в строку  $t$ . Во второй строке выведите в возрастающем порядке номера символов строки  $t$ , с которых начинаются вхождения  $p$ . Символы нумеруются с единицы.

### Пример

стандартный ввод	стандартный вывод
aaaa	4
Caaabdaaaa	1 2 6 7

## Задача I. Чтение строк

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

Дано дерево  $T$  состоящее из  $n$  вершин (пронумерованных целыми числами от 1 до  $n$ ). В каждой вершине записана некоторая буква. Корень дерева расположен в вершине 1.

Рассмотрим поддерево  $T_v$  некоторой вершины  $v$ . Вдоль любого просто пути, начинающегося в  $v$  и заканчивающегося в некоторой вершине  $u \in T_v$  (возможно, в самой  $v$ ), можно прочесть некоторую строку. Обозначим количество **различных** строк, которые можно прочесть таким способом как  $\text{dif}(v)$ .

Дополнительно: для каждой вершины  $v$  дано целое число  $c_v$ . Нас интересуют вершины, в которых значение  $\text{dif}(v) + c_v$  как можно больше.

Вы должны вычислить две величины — максимальное значение  $\text{dif}(v) + c_v$  и количество вершин  $v$  с максимальным  $\text{dif}(v) + c_v$ .

### Формат входных данных

В первой строке входных данных записано единственное целое число  $n$  ( $1 \leq n \leq 300\,000$ ) — количество вершин в дереве  $T$ .

Во второй строке записано  $n$  целых чисел  $c_i$  ( $0 \leq c_i \leq 10^9$ ).

В третьей строке записана строка  $s$ , состоящая из  $n$  строчных букв английского алфавита, —  $i$ -й символ этой строки соответствует букве, записанной в вершине  $i$ .

Далее следует  $n - 1$  строка с описанием рёбер дерева  $T$ . Каждая из них содержит два целых числа  $u$  и  $v$  ( $1 \leq u, v \leq n$ ), обозначающих ребро, которое соединяет вершины  $u$  и  $v$ .

Гарантируется, что входные данные описывают дерево.

### Формат выходных данных

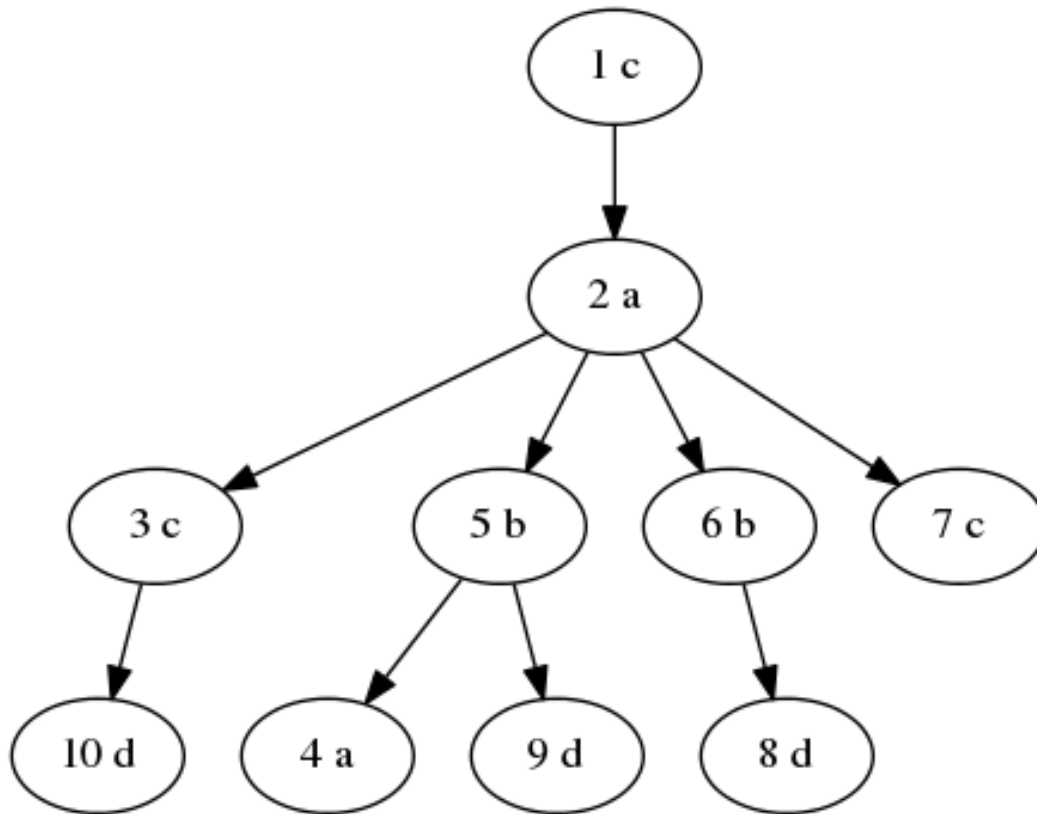
Выведите два числа — значение  $m = \max(\text{dif}(i) + c_i)$  для всех  $1 \leq i \leq n$  и количество вершин  $v$ , для которых  $m = \text{dif}(v) + c_v$ .

### Примеры

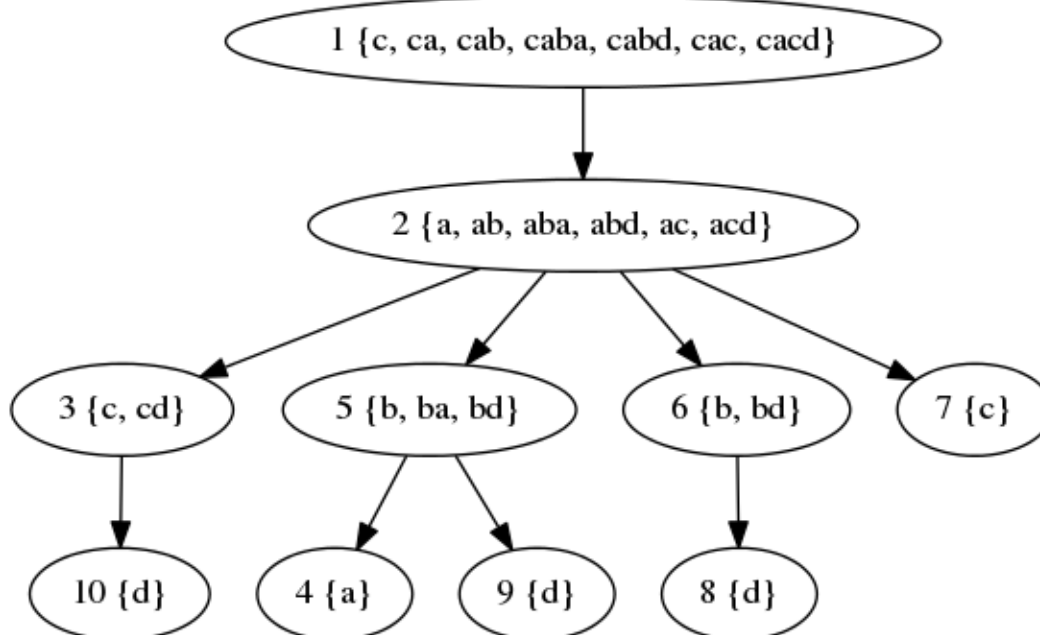
стандартный ввод	стандартный вывод
10 1 2 7 20 20 30 40 50 50 50 cacabbcddd 1 2 6 8 7 2 6 2 5 4 5 9 3 10 2 5 2 3	51 3
6 0 2 4 1 1 1 raaaba 1 2 2 3 2 4 2 5 3 6	6 2

## Замечание

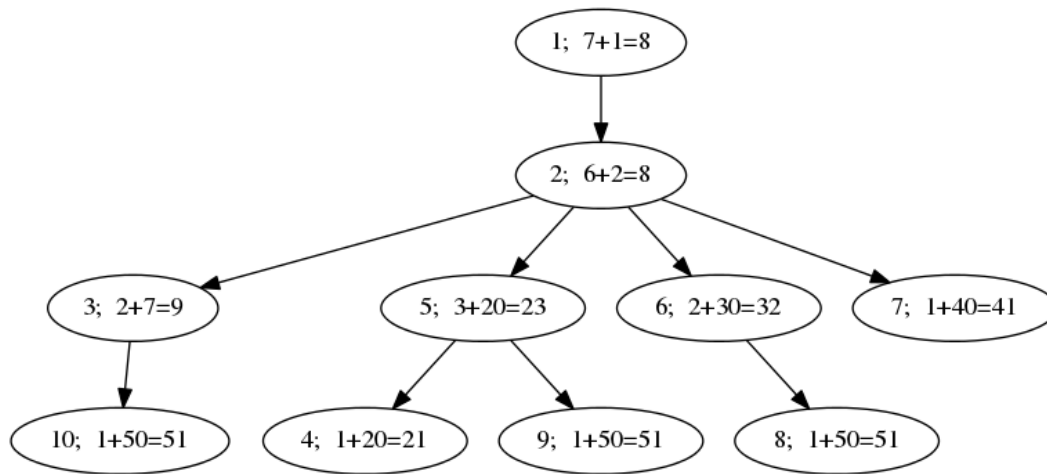
В первом примере дерево выглядит следующим образом:



Наборы строк, которые могут быть прочитаны из вершин:



Наконец, значения  $\text{dif}(v) + c_v$  таковы:



Во втором примере значения  $\text{dif}(1..n)$  таковы:  $(5, 4, 2, 1, 1, 1)$ . Различные строки, которые можно прочитать из вершины 2 таковы:  $a, aa, aaa, ab$ ; обратите внимание, что  $aa$  может быть прочитано как на пути до вершины 3, так и на пути до вершины 4.

## Задача J. Вася и Циклические Сдвиги

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

Однажды Василий проголодался и пошел на кухню подкрепиться. Но каково было его удивление, когда, открыв холодильник, он обнаружил там не привычные продукты, а строку! Причем не просто строку, а заикленную, прямо как бублик (ассоциации с едой часто приходят Васе в голову, когда он голоден). Загадочная строка заинтересовала любопытного, но все еще голодного Василия, и он начал ее вертеть в руках. Например, если вертеть строку `abacaba`, то можно получить следующие строки:

- `abacaba`
- `bacabaa`
- `acabaab`
- `cabaaba`
- `abaabac`
- `baabaca`
- `aabacab`

И тут Василия осенило: если он сможет посчитать, сколько раз в процессе кручения строки получается лексикографически минимальная строка, то еда магическим образом появится в холодильнике (странные идеи часто приходят Васе в голову, когда он голоден). Помогите Васе, иначе он так и будет сидеть голодным.

Более формально, вам дана строка. *Циклическим сдвигом* строки  $s$  длины  $n$  называется строка, полученная из исходной путем отбрасывания первых  $0 \leq k < n$  символов и приписывания их в конец. Необходимо посчитать, сколько раз среди всех циклических сдвигов строки встречается лексикографически минимальный циклический сдвиг.

### Формат входных данных

Единственная строка входного файла содержит строку  $S$ , найденную Василием. Она непуста, состоит из маленьких латинских букв, и её длина не превосходит 10 000 000.

### Формат выходных данных

Выведите единственное число — искомое количество минимальных циклических сдвигов.

### Примеры

стандартный ввод	стандартный вывод
<code>aaaa</code>	4
<code>abacaba</code>	1

### Замечание

В первом примере минимальным циклическим сдвигом является строка `aaaa`. Во втором примере минимальным циклическим сдвигом является строка `aabacab`.

## Задача К. Вирусы

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

Комитет По Исследованию Бинарных Вирусов обнаружил, что некоторые последовательности единиц и нулей являются кодами вирусов. Комитет изолировал набор кодов вирусов. Последовательность из единиц и нулей называется безопасной, если никакой ее подотрезок (т.е. последовательность из соседних элементов) не является кодом вируса. Сейчас цель комитета состоит в том, чтобы установить, существует ли бесконечная безопасная последовательность из единиц и нулей.

### Формат входных данных

Первая строка входного файла содержит одно целое число  $N$ , равное количеству всех вирусных кодов. Каждая из следующих  $n$  строк содержит непустое слово, составленное из символов 0 и 1 — код вируса. Суммарная длина всех слов не превосходит 30000.

### Формат выходных данных

Первая и единственная строка выходного файла должна содержать слово:

- ТАК — если бесконечная, безопасная последовательность из нулей и единиц существует;
- NIE — в противном случае.

### Примеры

стандартный ввод	стандартный вывод
3 01 11 00000	NIE
3 011 11 0000	ТАК

## Задача L. Под-бор

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

Бором называется подвешенное дерево, на каждом из рёбер которого написано по символу, причём символы, написанные на рёбрах, выходящих из общей вершины-родителя, различны. Будем называть направление от родителя к детям “вниз”. Назовем вхождением строки  $s$  в бор такую вершину бора, от которой можно пройти несколько шагов вниз таким образом, что встретившиеся символы образуют строку  $s$ .

Даны бор и несколько строк, найдите сумму количеств вхождений этих строк в этот бор.

### Формат входных данных

В первой строке входного файла записано единственное число  $n$ ,  $1 \leq n \leq 100\,000$  — количество вершин бора. В следующих  $n$  строках описаны вершины бора. В  $(i + 1)$ -й строке описаны дети  $i$ -й вершины: число  $k_i$  ее детей, затем  $k_i$  пар из номера вершины-ребёнка и символа, написанного на соответствующем ребре. Номер родителя всегда меньше номера ребёнка; корнем бора является вершина номер 1.

В  $(n + 2)$ -й строке записано количество  $m$  ( $1 \leq m \leq 100\,000$ ) строк для поиска. В следующих  $m$  строках перечислены сами строки. Входные строки непусты, а их суммарная длина не превышает  $100\,000$  символов.

Все символы, написанные на рёбрах, а также все символы, составляющие строки — маленькие латинские буквы.

### Формат выходных данных

Выведите одно число — сумму количеств вхождений.

### Пример

стандартный ввод	стандартный вывод
7	9
2 2 a 4 b	
2 3 a 6 b	
0	
1 5 b	
1 7 b	
0	
0	
4	
b	
bb	
bbb	
bb	