

Задача А. Неточное совпадение

Имя входного файла: `inexact-matching.in`
Имя выходного файла: `inexact-matching.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

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 входного файла

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

Формат выходного файла

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.

Примеры

<code>inexact-matching.in</code>	<code>inexact-matching.out</code>
<code>aaaa</code>	<code>4</code>
<code>Caaabdaaaa</code>	<code>1 2 6 7</code>

Задача В. Много строк

Имя входного файла: `musubstr.in`
Имя выходного файла: `musubstr.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Даны K строк из маленьких латинских букв. Требуется найти их наибольшую общую подстроку.

Формат входного файла

В первой строке число K ($1 \leq K \leq 10$). В следующих K строках — собственно K строк (длины строк от 1 до 10 000).

Формат выходного файла

Наибольшая общая подстрока.

Примеры

<code>musubstr.in</code>	<code>musubstr.out</code>
<code>3</code>	<code>cab</code>
<code>abacaba</code>	
<code>mycabarchive</code>	
<code>acabistrue</code>	

Задача С. Палиндромы

Имя входного файла: `palindr.in`
Имя выходного файла: `palindr.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Строка называется палиндромом, если она одинаково читается как слева направо, так и справа налево. Например, „abba“ — палиндром, а „omax“ — нет. Для строки α будем обозначать $\alpha[i..j]$ ее подстроку длины $j - i + 1$ с i -й по j -ю позицию включительно (позиции нумеруются с 1). Для заданной строки α длины N ($1 \leq N \leq 100\,000$) требуется подсчитать число q пар (i, j) , $1 \leq i < j \leq n$, таких что $\alpha[i..j]$ является палиндромом.

Формат входного файла

Входной файл содержит одну строку α длины N , состоящую из маленьких латинских букв.

Формат выходного файла

В выходной файл выведите искомое число q .

Примеры

<code>palindr.in</code>	<code>palindr.out</code>
<code>aaa</code>	<code>3</code>
<code>abba</code>	<code>2</code>
<code>omax</code>	<code>0</code>

Задача D. Яндекс

Имя входного файла: `yandex.in`
Имя выходного файла: `yandex.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Яндекс работает в одной очень известной компании. Его работа не очень сложна, но требует много времени. В основном Яндекс ищет некоторые данные в одной книге и переписывает их в другую. Яндекса не особенно волнует, кому нужны результаты его работы, главное — что за работу хорошо платят. Яндекс пришел в эту компанию не так давно, поэтому он трудится добросовестно и очень устает к концу дня. К концу дня для него все символы в книге сливаются, так что все эти ценные данные — это одна длинная строка, но он должен еще работать и работать с ними дальше и дальше... Может, босс заметит, как тщательно работает Яндекс, и повысит его...

Но... О, нет... Пока Яндекс мечтал, он забыл, что он должен был посмотреть в первой книге... После перерыва и чашки чая «Липтон» он кое-что вспомнил. В-первых, он вспомнил, что он должен был искать какую-то строку в первой книге. Во-вторых, он вспомнил, что во вторую книгу он должен был выписывать позиции, в которых встречалась эта строка, и что он уже выписал их все.

Формат входного файла

Во входном файле содержатся несколько тестов. Описание каждого теста начинается с натуральных чисел n ($1 \leq n \leq 1\,000\,000$) — количество символов в первой книге — и k ($1 \leq k \leq n$) — количество позиций, в которых Яндекс уже нашел вхождения искомой строки в текст (т. е. количество чисел во второй книге). На второй строке описания теста находится текст из первой книги — последовательность символов с ASCII-кодами, большими, чем 64. Третья строка описания теста содержит k номеров позиций, которые были записаны во второй книге.

Строка с $n = k = 0$ обозначает конец тестов; этот тест и все данные после него не должны быть обработаны.

Формат выходного файла

Для каждого теста выведите в выходной файл одну строку. Если существует строка, которая входит в текст в тех и только тех позициях, что указаны во второй книге, выведите одну строку „Correct. Length = x .. y .“, где x и y — минимально и максимально возможные длины искомой строки. Если решения не существует, выведите в выходной файл одну строку „Mistake.“.

Пример

<code>yandex.in</code>	<code>yandex.out</code>
5 2	Correct. Length = 2..3.
ababa	Correct. Length = 1..1.
1 3	Mistake.
1 1	
a	
1	
2 2	
ab	
1 2	
0 0	