

Задача А. Тандемный повтор

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Требуется проверить, является ли строка тандемным повтором, то есть является ли конкатенацией двух одинаковых строк.

Формат входных данных

На вход подается строка из строчных латинских букв длины не более 1000 символов

Формат выходных данных

В единственной строке выведите «YES», без кавычек, если строка является тандемным повтором, иначе выведите «NO», так же без кавычек.

Примеры

стандартный ввод	стандартный вывод
aaaa	YES
abcde	NO
abba	NO

Задача В. Слова не пройдут

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Дети, как известно, все раньше и раньше начинают пользоваться интернетом. Теперь, когда у них возникают вопросы, они не бегут к родителям, а заходят в свою любимую поисковую систему и узнают ответ в интернете. Но вдруг они случайно найдут что-нибудь, что им знать пока рановато? Или, может быть, лучше не знать вообще никогда?

В одной стране эту проблему решили очень просто: был создан список запрещенных для использования в интернете слов. Ведь очевидно, что статья, в которой упоминается какое-нибудь нехорошее слово, не может научить ребенка ничему хорошему. Любой сайт, содержащий хотя бы одно слово из этого списка, теперь подлежит мгновенной блокировке. Невинный ребенок никогда не натолкнется на что-нибудь, про что ему еще рановато знать — такой статьи просто не найдется в интернете. Но злобные сайтовладельцы придумали способ обойти этот запрет: если вместо некоторых букв написать внешне похожие на них цифры, то прочитав этот текст все равно будет можно, а робот, проверяющий сайты на пригодность, не распознает в слове запрещенное — ведь формально его нет на сайте.

Ваша задача — помочь правительству этой страны защитить детей от вредной информации. Напишите программу, которая будет проверять, нет ли в данной строке запрещенного слова, учитывая возможное коварство сайтовладельцев. Известно, что сайтовладельцы иногда делают следующие замены: $e \Rightarrow 3$, $o \Rightarrow 0$, $i \Rightarrow 1$, $t \Rightarrow 7$, $a \Rightarrow 4$, $s \Rightarrow 5$.

Формат входных данных

В первой строке входных данных дана строка — текст с сайта. Во второй строке входных данных дана другая строка — запрещенное слово. Первая строка состоит из маленьких латинских букв и цифр, вторая строка состоит только из маленьких латинских букв. Длина каждой строки не превышает 100.

Формат выходных данных

Выведите «YES», если запрещенное слово встречается как подстрока в строке с сайта, и «NO» иначе. Возможно, в строке с сайта некоторые буквы изначально были заменены на цифры в соответствии с приведенными выше правилами.

Примеры

стандартный ввод	стандартный вывод
inah0leinthegroundthereliv3dah0bb1t hobbit	YES
whath4v3igotinmypocket handses	NO
whath4veig0t1nmy0ck37 knife	NO
wh4thav31go71nmy0ck3t stringofnothing	NO

Задача С. Префикс-функция

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

Постройте префикс-функцию для заданной строки s .

Формат входных данных

Первая строка входного файла содержит s ($1 \leq |s| \leq 10^6$). Строка состоит из букв латинского алфавита.

Формат выходных данных

Выведите значения префикс-функции строки s для всех индексов $1, 2, \dots, |s|$.

Пример

<code>prefix.in</code>	<code>prefix.out</code>
aaaAAA	0 1 2 0 0 0

Задача D. Z-функция

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

Постройте Z-функцию для заданной строки s .

Формат входных данных

Первая строка входного файла содержит s ($1 \leq |s| \leq 10^6$). Строка состоит из букв латинского алфавита.

Формат выходных данных

Выведите значения Z-функции строки s для индексов $2, 3, \dots, |s|$.

Примеры

<code>z.in</code>	<code>z.out</code>
aaaAAA	2 1 0 0 0
abacaba	0 1 0 3 0 1

Задача Е. Кубики

Имя входного файла: `cubes.in`
Имя выходного файла: `cubes.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Привидение Петя любит играть со своими кубиками. Он любит выкладывать их в ряд и разглядывать свое 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 нестоящие, а какие — всего лишь отражение в зеркале. Помогите Пете! Выясните, сколько кубиков может быть у Пети. 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 число 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

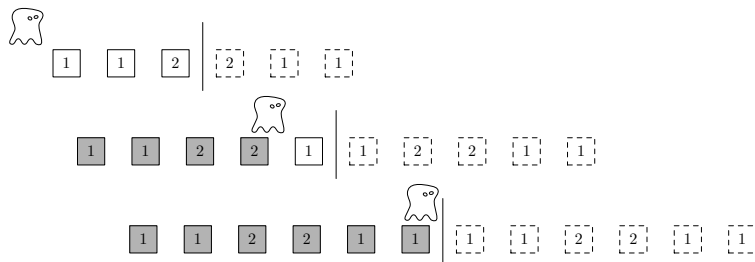
Выведите в выходной файл все такие K , что у Пети может быть K кубиков.

Пример

<code>cubes.in</code>	<code>cubes.out</code>
6 2 1 1 2 2 1 1	6 5 3

Замечание

В приведенном примере взаимные расположения Пети, кубиков и зеркала приведены на рисунке. Петя смотрит вправо, затененные на рисунке кубики находятся позади Пети и поэтому он их не видит.



Задача F. Подстроки-3

Имя входного файла: `substr3.in`
Имя выходного файла: `substr3.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

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

Формат входных данных

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 выходных данных

Наибольшая общая подстрока.

Пример

substr3.in	substr3.out
3 abacaba mycabarchive acabistrue	cab

Задача G. Помогите, спасите!

Имя входного файла: `keepcounted.in`
Имя выходного файла: `keepcounted.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Дана строка. Найдите для каждого её префикса количество различных подстрок в нём.

Формат входных данных

В единственной строке входных данных содержится непустая строка S , состоящая из N ($1 \leq N \leq 10^4$) маленьких букв английского алфавита.

Формат выходных данных

Выведите N строк, в i -й строке должно содержаться количество различных подстрок в i -м префиксе строки S .

Примеры

<code>keepcounted.in</code>	<code>keepcounted.out</code>
<code>aabab</code>	1
	2
	5
	8
	11
<code>atari</code>	1
	3
	5
	9
	14

Задача Н. Ретростроки

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 0.5 секунд
Ограничение по памяти: 256 мегабайт

Строкой S называется последовательность символов S_1, \dots, S_n , где $|S| = n$ — это *длина* строки S .

Для любого k ($1 \leq k \leq |S|$) k -м *префиксом* строки S называется строка S_1, \dots, S_k длины k . Если $k < |S|$, то префикс называется *собственным*.

Аналогично для любого k ($1 \leq k \leq |S|$) k -м *суффиксом* строки S называется строка $S_{|S|-k+1}, \dots, S_{|S|}$ длины k . Если $k < |S|$, то суффикс также называется *собственным*.

Назовём *числом повторяемости* строки S количество её различных собственных суффиксов, каждый из которых совпадает с префиксом той же длины, что и этот суффикс.

Назовём строку *ретрострокой*, если её число повторяемости строго больше чисел повторяемости всех её собственных префиксов.

Дана строка S . Нужно найти её префикс максимальной длины (не обязательно собственный), являющийся ретрострокой.

Формат входных данных

В первой строке входного файла записана строка S , $1 \leq |S| \leq 1000000$. Строка содержит лишь символы с ASCII-кодом от 33 до 126.

Формат выходных данных

В первой строке выходного файла должен быть выведен префикс S максимальной длины, являющийся ретрострокой.

Примеры

стандартный ввод	стандартный вывод
z	z
aabaabaabaabaabaaba	aabaabaabaaba

Задача I. Поиск периода

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

Дана строка s . Требуется найти минимальную по длине строку t , такую что s представима в виде конкатенации одной или нескольких строк t .

Формат входных данных

Первая строка входного файла содержит s ($1 \leq |s| \leq 10^6$). Строка состоит из букв латинского алфавита.

Формат выходных данных

Выведите длину искомой строки t .

Примеры

<code>period.in</code>	<code>period.out</code>
abcbcabcb	3
abacaba	7

Задача J. Словарь

Имя входного файла: `dictionary.in`
Имя выходного файла: `dictionary.out`
Ограничение по времени: 2 секунды
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 (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.

Пример

<code>dictionary.in</code>	<code>dictionary.out</code>
<code>trololo</code>	<code>No</code>
<code>3</code>	<code>Yes</code>
<code>abacabadabacaba</code>	<code>Yes</code>
<code>olo</code>	
<code>trol</code>	

Задача К. Быстрый поиск подстроки в строке

Имя входного файла: `search2.in`
Имя выходного файла: `search2.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.

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

<code>search2.in</code>	<code>search2.out</code>
<code>aba</code>	2
<code>abaCaba</code>	1 5

Задача L. Сравнения подстрок

Имя входного файла: `substrcmp.in`
Имя выходного файла: `substrcmp.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Дана строка. Нужно уметь отвечать на запросы вида: равны ли подстроки $[a..b]$ и $[c..d]$.

Формат входных данных

Сперва строка S (не более 10^5 строчных латинских букв). Далее число M — количество запросов.

В следующих M строках запросы a, b, c, d . $0 \leq M \leq 10^5, 1 \leq a \leq b \leq |S|, 1 \leq c \leq d \leq |S|$

Формат выходных данных

M строк. Выведите **Yes**, если подстроки совпадают, и **No** иначе.

Пример

substrcmp.in	substrcmp.out
trololo	Yes
3	Yes
1 7 1 7	No
3 5 5 7	
1 1 1 5	

Задача M. String Game

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

Любимое занятие Чебурашки — вычеркивать буквы из слова, чтобы получилось другое слово. Но получается это у него довольно плохо, потому что он еще маленький, да и вообще его никто не учил писать и читать. Поэтому ему всегда помогает его лучший друг Гена.

Гена дал Чебурашке слово t и хочет, чтобы из него получилось слово p . Чебурашка начинает вычеркивать буквы в некотором порядке, который задан перестановкой номеров букв слова t : $a_1 \dots a_{|t|}$. Заметим, что после вычеркивания буквы нумерация не меняется. Этот порядок изначально известен Гене. Задача крокодила Гены состоит в том, чтобы в некоторый момент времени остановить друга и закончить вычеркивание самому, получив после этого слово p . Так как Чебурашке нравится это занятие, Гена хочет остановить его как можно позже. Ваша задача — сообщить, сколько букв может вычеркнуть Чебурашка до того, как его остановит Гена.

Гарантируется, что слово p можно получить вычеркиванием букв из t .

Формат входных данных

Первая и вторая строки входного файла содержат слова t и p , соответственно. Слова состоят из строчных букв латинского алфавита ($1 \leq |p| < |t| \leq 200\,000$).

Следующая строка содержит перестановку $a_1 \dots a_{|t|}$ номеров букв, задающую порядок, в котором Чебурашка вычеркивает буквы слова t ($1 \leq a_i \leq |t|$, все a_i различны).

Формат выходных данных

Выведите одно число — максимальное число букв, которые может вычеркнуть Чебурашка.

Пример

стандартный ввод	стандартный вывод
ababcba abb 5 3 4 1 7 6 2	3

Пояснение к примеру

Последовательность вычеркивания букв Чебурашкой выглядит так:

«ababcba» → «abab**с**ba» → «abab**с****б**a» → «abab**с****б****с**ba»

Продолжать вычеркивать Чебурашка не может, потому что из «abab**с****б****с**ba» нельзя получить «abb».

Таким образом, Чебурашка может вычеркнуть только три буквы в своей последовательности.