

Задача А. Суффиксное дерево

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

Постройте суффиксное дерево для заданной строки s .

Формат входных данных

Первая строка входного файла содержит строку s ($1 \leq |s| \leq 100\,000$). Строка состоит из строчных латинских букв.

Формат выходных данных

В первой строке выходного файла выведите два натуральных числа n и m , разделенных пробелом — число вершин и ребер в суффиксном дереве соответственно. В следующих m строках выведите описания ребер в формате $\langle \text{родитель} \rangle \langle \text{потомок} \rangle \langle l \rangle \langle r \rangle$. Эта запись означает, что на ребре написана строка $s[l..r]$, при этом значение l должно быть минимально возможным. Корнем дерева должна быть вершина с номером 1. Вершины должны быть занумерованы натуральными числами, не превышающими n .

Примеры

suffixtree.in	suffixtree.out
ababb	7 6 1 4 1 2 1 6 2 2 4 2 3 5 4 5 5 5 6 3 3 5 6 7 5 5

Задача В. Ненокку

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

Очень известный автор не менее известной книги решил написать продолжение своего произведения. Он писал все свои книги на компьютере, подключенном к интернету. Из-за такой неосторожности мальчику Ненокку удалось получить доступ к еще ненаписанной книге. Каждый вечер мальчик залазил на компьютер писателя и записывал на свой компьютер новые записи. Ненокку, записав на свой компьютер очередную главу, заинтересовался, а использовал ли хоть раз писатель слово “книга”. Но он не любит читать книги (он лучше ползает в интернете), и поэтому он просит вас узнать есть ли то или иное слово в тексте произведения. Но естественно его интересует не только одно слово, а достаточно много.

Формат входных данных

В каждой строчке входного файла записано одна из двух записей.

1. ? <слово> (<слово> - это набор не более 50 латинских символов);
2. A <текст> (<текст> - это набор не более 10^5 латинских символов).

1 означает просьбу проверить существование подстроки <слово> в произведение.

2 означает добавление в произведение <текст>.

Писатель только начал работать над произведением, поэтому он не мог написать более 10^5 символов. А входной файл содержит не более 15 мегабайт информации.

Формат выходных данных

Выведите на каждую строчку типа 1 “YES”, если существует подстрока <слово>, и “NO” в противном случае. Не следует различать регистр букв.

Примеры

<code>nenokku.in</code>	<code>nenokku.out</code>
? love	NO
? is	NO
A Loveis	YES
? love	NO
? WHO	YES
A Whoareyou	
? is	

Задача С. Помогите, спасите!

Имя входного файла: `keepcounted.in`
Имя выходного файла: `keepcounted.out`
Ограничение по времени: 4 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 ($1 \leq N \leq 2 \cdot 10^5$) 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