

12 - Суффиксное дерево

A. Суффиксное дерево

2 секунды, 256 мегабайт

{256 мегабайт}

Постройте суффиксное дерево для заданной строки  $s$ .

Входные данные

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

Выходные данные

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  $n$ .

входные данные
ababb
выходные данные
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

B. Количество подстрок

2 секунды, 512 мегабайт

Вычислите количество различных подстрок строки  $s$ .

Входные данные

Единственная строка входного файла содержит строку  $s$  ( $1 \leq |s| \leq 400\,000$ ). Строка состоит из строчных латинских букв.

Выходные данные

Выведите одно число — ответ на задачу.

входные данные
ababb
выходные данные
11

C. Множественный поиск 2

3 секунды, 1024 мегабайта

Дан массив строк  $s_i$  и строка  $t$ . Требуется для каждой строки  $s_i$  определить, сколько раз она встречается в  $t$  как подстрока.

Входные данные

Первая строка входного файла содержит целое число  $n$  — число элементов в  $s$  ( $1 \leq n \leq 10^6$ ). Следующие  $n$  строк содержат по одной строке  $s_i$ . Сумма длин всех строк из  $s$  не превосходит  $10^6$ . Последняя строка входного файла содержит  $t$  ( $1 \leq t \leq 10^6$ ). Все строки состоят из строчных латинских букв.

Выходные данные

For each line of  $s_i$  print one number: how many times it occurs in  $t$ . Lines are numbered in the order they appear in the input file.

входные данные
3 abc abcd abcde abcdef
выходные данные
1 0 1

D. Ненокку

2 секунды, 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.

Входные данные

В каждой строчке входного файла записано одна из двух записей.

- 1. ?  $\langle \text{слово} \rangle$  ( $\langle \text{слово} \rangle$  - это набор не более 50 латинских символов);
- 2. A  $\langle \text{текст} \rangle$  ( $\langle \text{текст} \rangle$  - это набор не более  $10^5$  латинских символов).

1 означает просьбу проверить существование подстроки  $\langle \text{слово} \rangle$  в произведение.

2 означает добавление в произведение  $\langle \text{текст} \rangle$ .

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

Выходные данные

Выведите на каждую строчку типа 1 "YES", если существует подстрока  $\langle \text{слово} \rangle$ , и "NO" в противном случае. Не следует различать регистр букв.

входные данные
? love ? is A Loveis ? love ? WHO A Whoareyou ? is
выходные данные
NO NO YES NO YES

E. Суффиксный массив

2 секунды, 512 мегабайт

Постройте суффиксный массив для заданной строки  $s$ , для каждых двух соседних суффиксов найдите длину максимального общего префикса.

Входные данные

Первая строка входного файла содержит строку  $s$  ( $1 \leq |s| \leq 400\,000$ ).  
Строка состоит из строчных латинских букв.

**Выходные данные**

В первой строке выведите  $|s|$  различных чисел — номера первых символов суффиксов строки  $s$  так, чтобы соответствующие суффиксы были упорядочены в лексикографически возрастающем порядке. Во второй строке выведите  $|s| - 1$  чисел — длины наибольших общих префиксов.

входные данные
ababb
выходные данные
1 3 5 2 4 2 0 1 1