

## Задача А. Роскомнадзор-1

Имя входного файла: `input.txt`  
Имя выходного файла: `output.txt`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Реализуйте алгоритм, который заменяет все вхождения строки  $s$  в строке  $t$  на соответствующее количество символов «\*». Пробельные символы оставьте нетронутыми. Поиск осуществляйте регистронезависимый, считайте символы перевода строки и пробела равными.

Алгоритм должен гарантированно правильно работать за время  $O(|t| + |s|)$ , используя  $O(|t|)$  памяти.

Ревью этой задачи будет производиться с нулевой толерантностью к копиям в коде.

### Формат входных данных

В первой строке входных данных дана строка  $s$ , состоящая из букв латинского алфавита, цифр и пробелов. В следующих строках дана строка  $t$ , состоящая из букв латинского алфавита, цифр, пробелов и переводов строк.

### Формат выходных данных

Выведите строку  $t$  после применения цензуры.

### Пример

input.txt	output.txt
tor vpn Torrent test123 tor vpn torrent tor vpn torrent contest  NaGiBaToR vpn torrenTor vpn torrentsru	

## Задача В. Роскомнадзор-2

Имя входного файла: `input.txt`  
 Имя выходного файла: `output.txt`  
 Ограничение по времени: 1 секунда  
 Ограничение по памяти: 256 мегабайт

Реализуйте алгоритм, который заменяет все вхождения строк  $s_1, s_2, \dots$  в строке  $t$  на соответствующее количество симфолов «\*». Пробельные символы оставьте нетронутыми. Поиск осуществляйте регистронезависимый, считайте символы перевода строки и пробела равными.

После каждой позиции, в которой окончилось вхождение, в фигурных скобках выведите в возрастающем порядке номера строк, вхождение которых оканчивается в этой позиции. Номера разделите запятыми.

Алгоритм должен гарантированно правильно работать за время  $O(|t| + \sum |s_i| + ans \cdot \log ans)$ , используя  $O(|t| + ans)$  памяти, где  $ans$  — суммарное количество вхождений.

Ревью этой задачи будет производиться с нулевой толерантностью к копиям в коде.

### Формат входных данных

В первых строках даны  $s_1, s_2, \dots$ , по одной в каждой строке. Множество строк  $s_i$  завершается пустой строкой. Строки нумеруются с единицы. Строки состоят из букв латинского алфавита, цифр и пробелов.

В следующих строках дана строка  $t$ , состоящая из букв латинского алфавита, цифр, пробелов и переводов строк.

### Формат выходных данных

Выведите строку  $t$  после применения цензуры с указанием всех вхождений.

### Пример

input.txt	output.txt
pupa	Prishli k*{4}k to ****{1,4} i
LUPa	****{2,4} **{3}luch*{4}t z*{4}rpl*{4}tu
pa po	No v buhg*{4}lterii vsjo
a	pereput*{4}li i ****{2,4}
	**{3}luchil z*{4}rpl*{4}tu z*{4} Pupu
Prishli kak to Pupa i	
Lupa poluchat zarplatu	
No v buhgalterii vsjo	
pereputali i Lupa	
poluchil zarplatu za Pupu	

## Задача С. Суффиксный массив

Имя входного файла: `input.txt`  
Имя выходного файла: `output.txt`  
Ограничение по времени: 10 секунд  
Ограничение по памяти: 256 мегабайт

Выведите индексы всех суффиксов строки, отсортированных в лексикографическом порядке. Например, у строки «АВВАВСАВ» индексы отсортированных индексов выглядят так:

- 8:
- 6:АВ
- 0:АВВАВСАВ
- 3:АВСАВ
- 7:В
- 2:ВАВСАВ
- 1:ВВАВСАВ
- 4:ВСАВ
- 5:САВ

Алгоритм должен гарантированно правильно работать за время  $O(|s| \log |s|)$ .

Ревью этой задачи будет производиться с нулевой толерантностью к копиям в коде.

### Формат входных данных

Одна строка  $s$ , состоящая из заглавных латинских символов.

### Формат выходных данных

Выведите  $|s| + 1$  число — искомую последовательность номеров суффиксов строки  $s$ .

### Примеры

input.txt	output.txt
АВВАВСАВ	8 6 0 3 7 2 1 4 5
МИССИССИПИ	11 10 7 4 1 0 9 8 6 3 5 2