

## ЗАДАЧА 1

Даны две строки:  $s_1$  и  $s_2$  с одинаковым размером, проверьте, может ли некоторая перестановка строки  $s_1$  “победить” некоторую перестановку строки  $s_2$  или наоборот.

Строка  $x$  может “победить” строку  $y$  (обе имеют размер  $n$ ), если  $x[i] \geq y[i]$  (в алфавитном порядке) для всех  $i$  от 0 до  $n-1$ .

Примеры:

```
Input: s1 = "abc", s2 = "xya"
```

```
Output: true
```

Объяснение: «аух» – это перестановка строки  $s_2 = \text{«хуа»}$ , которая “побеждает” строку  $s_1 = \text{«abc»}$ .

```
Input: s1 = "abe", s2 = "acd"
```

```
Output: false
```

Объяснение: Все перестановки для  $s_1 = \text{«abe»}$ : «abe», «aeb», «bae», «bea», «eab» и «eba», а все перестановки для  $s_2 = \text{«acd»}$ : «acd», «adc», «cad», «cda», «dac» и «ca». Однако нет никакой перестановки строки  $s_1$ , которая может нарушить некоторую перестановку строки  $s_2$  и наоборот.

```
s1.length == n
```

```
s2.length == n
```

```
1 <= n <= 10^5
```

## ЗАДАЧА 2

Дана строка  $s$ , вернуть самую длинную палиндромную подстроку в  $s$ .

Примеры:

Input:  $s = \text{"babad"}$

Output:  $\text{"bab"}$

Note:  $\text{"aba"}$  is also a valid answer.

Input:  $s = \text{"cbabd"}$

Output:  $\text{"bb"}$

### ЗАДАЧА 3

Вернуть количество отдельных непустых подстрок текста, которые могут быть записаны как конкатенация некоторой строки с самой собой (т.е. она может быть записана, как  $a + a$ , где  $a$  - некоторая строка).

Примеры:

Input: `text = "abcabcabc"`

Output: `3`

Explanation: The 3 substrings are "abcabc", "bcabca" and "cabcab".