

Problem 555: Split Concatenated Strings

Problem Information

Difficulty: Medium

Acceptance Rate: 43.47%

Paid Only: Yes

Tags: Array, String, Greedy

Problem Description

You are given an array of strings `strs`. You could concatenate these strings together into a loop, where for each string, you could choose to reverse it or not. Among all the possible loops

Return `_`the lexicographically largest string after cutting the loop, which will make the looped string into a regular one`_`.

Specifically, to find the lexicographically largest string, you need to experience two phases:

1. Concatenate all the strings into a loop, where you can reverse some strings or not and connect them in the same order as given. 2. Cut and make one breakpoint in any place of the loop, which will make the looped string into a regular one starting from the character at the cutpoint.

And your job is to find the lexicographically largest one among all the possible regular strings.

Example 1:

Input: `strs = ["abc", "xyz"]` **Output:** `"zyxcba"` **Explanation:** You can get the looped string `"-abcxyz-", "-abczyx-", "-cbaxyz-", "-cbazyx-"`, where `'-'` represents the looped status. The answer string came from the fourth looped one, where you could cut from the middle character `'a'` and get `"zyxcba"`.

Example 2:

Input: `strs = ["abc"]` **Output:** `"cba"`

****Constraints:****

* `1 <= strs.length <= 1000` * `1 <= strs[i].length <= 1000` * `1 <= sum(strs[i].length) <= 1000`
* `strs[i]` consists of lowercase English letters.

Code Snippets

C++:

```
class Solution {
public:
    string splitLoopedString(vector<string>& strs) {

    }
};
```

Java:

```
class Solution {
    public String splitLoopedString(String[] strs) {

    }
}
```

Python3:

```
class Solution:
    def splitLoopedString(self, strs: List[str]) -> str:
```