Problem 960: Delete Columns to Make Sorted III

Problem Information

Difficulty: Hard
Acceptance Rate: 59.48%
Paid Only: No
Tags: Array, String, Dynamic Programming

Problem Description

You are given an array of `n` strings `strs`, all of the same length.

We may choose any deletion indices, and we delete all the characters in those indices for each string.

For example, if we have `strs = ["abcdef","uvwxyz"]` and deletion indices `{0, 2, 3}`, then the final array after deletions is `["bef", "vyz"]`.

Suppose we chose a set of deletion indices `answer` such that after deletions, the final array has **every string (row) in lexicographic** order. (i.e., `(strs[0][0] <= strs[0][1] <= ... <= strs[0][strs[0].length - 1])`, and `(strs[1][0] <= strs[1][1] <= ... <= strs[1][strs[1].length - 1])`, and so on). Return _the minimum possible value of_ `answer.length`.

Example 1:

Input: strs = ["babca","bbazb"] **Output:** 3 **Explanation:** After deleting columns 0, 1, and 4, the final array is strs = ["bc", "az"]. Both these rows are individually in lexicographic order (ie. strs[0][0] <= strs[0][1] and strs[1][0] <= strs[1][1]). Note that strs[0] > strs[1] - the array strs is not necessarily in lexicographic order.

Example 2:

Input: strs = ["edcba"] **Output:** 4 **Explanation:** If we delete less than 4 columns, the only row will not be lexicographically sorted.

Example 3:

Input: strs = ["ghi","def","abc"] **Output:** 0 **Explanation:** All rows are already lexicographically sorted.

Constraints:

* `n == strs.length` * `1 <= n <= 100` * `1 <= strs[i].length <= 100` * `strs[i]` consists of lowercase English letters.

*

Code Snippets

C++:

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

}
};
```

Java:

```
class Solution {
public int minDeletionSize(String[] strs) {

}
}
```

Python3:

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