Problem 2381: Shifting Letters II

Problem Information

Difficulty: Medium
Acceptance Rate: 53.45%
Paid Only: No
Tags: Array, String, Prefix Sum

Problem Description

You are given a string `s` of lowercase English letters and a 2D integer array `shifts` where `shifts[i] = [starti, endi, directioni]`. For every `i`, **shift** the characters in `s` from the index `starti` to the index `endi` (**inclusive**) forward if `directioni = 1`, or shift the characters backward if `directioni = 0`.

Shifting a character **forward** means replacing it with the **next** letter in the alphabet (wrapping around so that `'z'` becomes `'a'`). Similarly, shifting a character **backward** means replacing it with the **previous** letter in the alphabet (wrapping around so that `'a'` becomes `'z'`).

Return _the final string after all such shifts to_`s` _are applied_.

Example 1:

Input: s = "abc", shifts = [[0,1,0],[1,2,1],[0,2,1]] **Output:** "ace" **Explanation:** Firstly, shift the characters from index 0 to index 1 backward. Now s = "zac". Secondly, shift the characters from index 1 to index 2 forward. Now s = "zbd". Finally, shift the characters from index 0 to index 2 forward. Now s = "ace".

Example 2:

Input: s = "dztz", shifts = [[0,0,0],[1,1,1]] **Output:** "catz" **Explanation:** Firstly, shift the characters from index 0 to index 0 backward. Now s = "cztz". Finally, shift the characters from index 1 to index 1 forward. Now s = "catz".

Constraints:

* `1 <= s.length, shifts.length <= 5 * 104` * `shifts[i].length == 3` * `0 <= starti <= endi < s.length` * `0 <= directioni <= 1` * `s` consists of lowercase English letters.

Code Snippets

C++:

```
class Solution {
public:
string shiftingLetters(string s, vector<vector<int>>& shifts) {

}
};
```

Java:

```
class Solution {
public String shiftingLetters(String s, int[][] shifts) {

}
}
```

Python3:

```
class Solution:
def shiftingLetters(self, s: str, shifts: List[List[int]]) -> str:
```