Problem 848: Shifting Letters

Problem Information

Difficulty: Medium
Acceptance Rate: 45.90%
Paid Only: No
Tags: Array, String, Prefix Sum

Problem Description

You are given a string `s` of lowercase English letters and an integer array `shifts` of the same length.

Call the `shift()` of a letter, the next letter in the alphabet, (wrapping around so that `'z'` becomes `'a'`).

* For example, `shift('a') = 'b'`, `shift('t') = 'u'`, and `shift('z') = 'a'`.

Now for each `shifts[i] = x`, we want to shift the first `i + 1` letters of `s`, `x` times.

Return _the final string after all such shifts to s are applied_.

Example 1:

Input: s = "abc", shifts = [3,5,9] **Output:** "rpl" **Explanation:** We start with "abc". After shifting the first 1 letters of s by 3, we have "dbc". After shifting the first 2 letters of s by 5, we have "igc". After shifting the first 3 letters of s by 9, we have "rpl", the answer.

Example 2:

Input: s = "aaa", shifts = [1,2,3] **Output:** "gfd"

Constraints:

* `1 <= s.length <= 105` * `s` consists of lowercase English letters. * `shifts.length == s.length` * `0 <= shifts[i] <= 109`

Code Snippets

C++:

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

}
};
```

Java:

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

}
}
```

Python3:

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