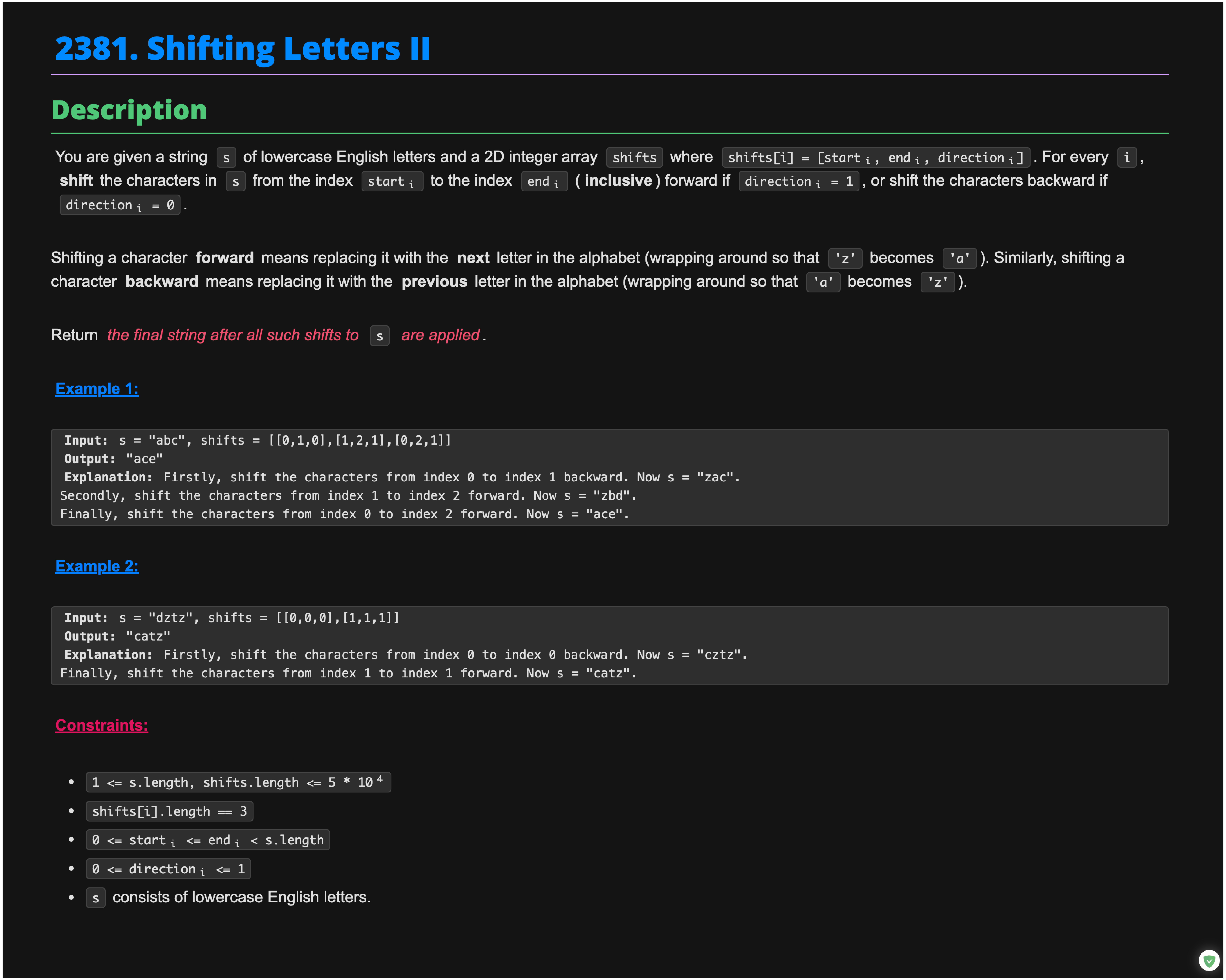

2381. Shifting Letters II

Description

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

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 <=` $5 * 10^4$
- `shifts[i].length == 3`
- $0 <= start_i <= end_i < s.length$
- $0 <= direction_i <= 1$
- `s` consists of lowercase English letters.