

**32. Perform String Shifts** You are given a string *s* containing lowercase English letters, and a matrix *shift*, where *shift[i] = [direction<sub>i</sub>, amount<sub>i</sub>]*: • *direction<sub>i</sub>* can be 0 (for left shift) or 1 (for right shift). • *amount<sub>i</sub>* is the amount by which string *s* is to be shifted. • A left shift by 1 means remove the first character of *s* and append it to the end. • Similarly, a right shift by 1 means remove the last character of *s* and add it to the beginning. Return the final string after all operations. Example 1: Input: *s* = "abc", *shift* = [[0,1], [1,2]] Output: "cab" Explanation: [0,1] means shift to left by 1. "abc" -> "bca" [1,2] means shift to right by 2. "bca" -> "cab"

**PROGRAM:**

```
def string_shift(s, shift):
    total_shift = 0
    for direction, amount in shift:
        if direction == 0:
            total_shift -= amount
        else:
            total_shift += amount
    total_shift %= len(s)
    return s[-total_shift:] + s[:-total_shift]

s = "abcdefg"
shift = [[1,1], [1,1], [0,2], [1,3]]
result = string_shift(s, shift)
print(result)
```

**OUTPUT:**

```
PS C:\Users\chall\OneDrive\Desktop\DAA> & C:/Users/chall/AppData/Local/Programs/Python/Python312/python.exe
"
efgabcd
PS C:\Users\chall\OneDrive\Desktop\DAA>
```

**TIME COMPLEXITY:**

**Time complexity for the above code is**

$$\mathbf{F(n)=O(m+n)}$$