

32)Perform String Shifts You are given a string s containing lowercase English letters, and a matrix shift, where $\text{shift}[i] = [\text{direction}_i, \text{amount}_i]$:

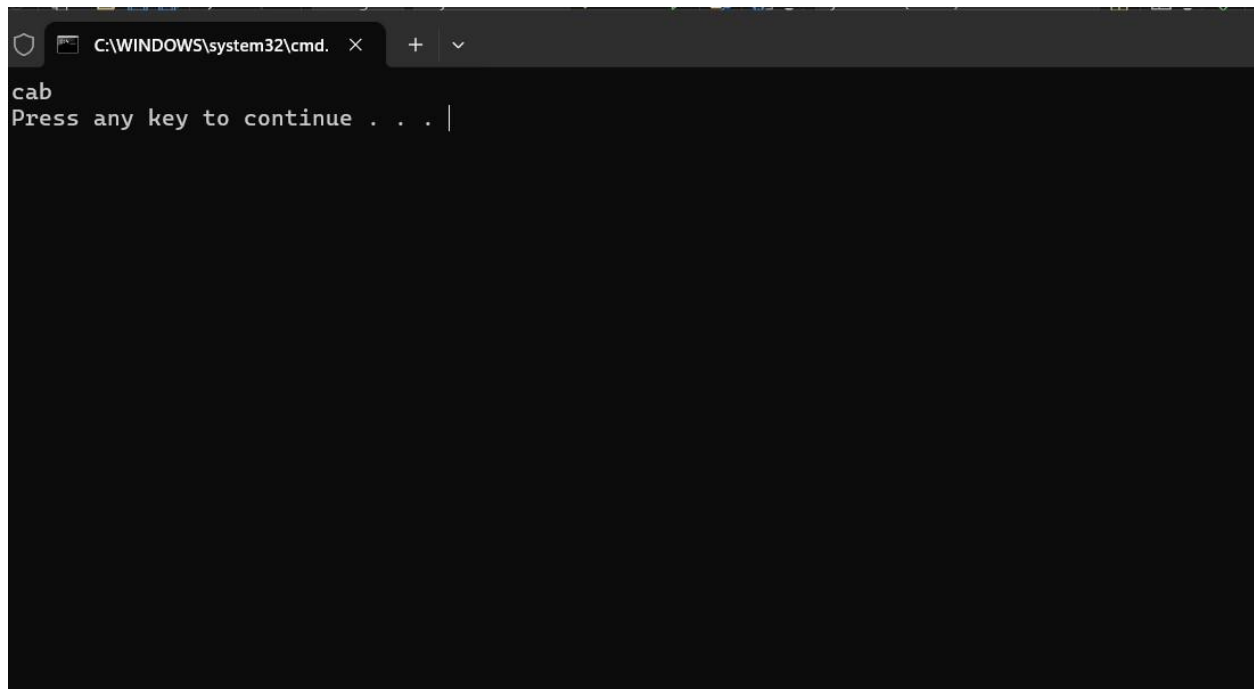
- direction_i can be 0 (for left shift) or 1 (for right shift).
- amount_i is the amount by which string s is to be shifted.
- A left shift by 1 means remove the first character of s and append it to the end.
- Similarly, a right shift by 1 means remove the last character of s and add it to the beginning.

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"

CODE:

```
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:

A screenshot of a Windows command prompt window. The title bar shows the path 'C:\WINDOWS\system32\cmd.' and standard window controls. The command prompt displays the output 'cab' on the first line and 'Press any key to continue . . . |' on the second line, with a vertical cursor bar at the end of the second line.

TIME COMPLEXITY : $O(m+n)$