Products of array except self - prefix and suffix

```
class Solution:
    def prod_of_array(self, nums: List[int]) -> List[int]:

        n = len(nums)
        result = [0] * n
        pref = [0] * n
        suff = [0] * n

        pref[0] = suff[n-1] = 1
        for i in range(1, n):
            pref[i] = nums[i-1] * pref[i-1]
        for i in range(n-2, -1, -1):
            suff[i] = nums[i+1] * suff[i+1]
        for i in range(n):
            result[i] = pref[i] * suff[i]
        return result

print(Solution().prod_of_array(nums = [1,2,4,6]))
```

Step 1:

self → solution instance

nums → [1, 2, 4, 6]

n → 4

result → [0, 0, 0, 0]

pref → [0, 0, 0, 0]

suff → [0, 0, 0, 0]

Step 2: pref[0] = suff[n-1] = 1

pref → [1, 0, 0, 0]

suff → [0, 0, 0, 1]

Step 3: for i in range(1, n):

pref[i] = nums[i-1] * pref[i-1]

Iterations:

1st. i → 1

pref[1] = nums[0] * pref[0] => 1

pref → [1, 1, 0, 0]

2nd. i → 2

pref[2] = nums[1] * pref[1] => 2

pref -> [1,1,2,0]

3rd. i -> 3

pref[3] = nums[2] * pref[2] => 8

pref -> [1,1,2,8]

Step 4: for i in range(n-2, -1, -1):

suff[i] = nums[i+1] * suff[i+1]

start with (4-2)=2, keep going until you get to just before -1, by steps of -1 (reverse).

Iterations:

1st. i -> 2

suff[2] = nums[3] * suff[3] => 6

suff -> [0,0,6,1]

2nd. i -> 1

suff[1] = nums[2] * suff[2] => 24

suff -> [0,24,6,1]

3rd. $i \to 0$

$suff[0] = nums[1] * suff[1] \Rightarrow 48$

$suff \to [48, 24, 6, 1]$

Step 5:
```
for i in range(n):
    result[i] = pref[i] * suff[i]
```

pref [1, 1, 2, 8]

suff [48, 24, 6, 1]

Iterations:

1st. $i \to 0$

$result[0] = pref[0] * suff[0] \Rightarrow 48$

$result \to [48, 0, 0, 0]$

2nd. $i \to 1$

$result[1] = pref[1] * suff[1] \Rightarrow 24$

$result \to [48, 24, 0, 0]$

3rd. $i \to 2$

$result[2] = pref[2] * suff[2] \Rightarrow 12$

$result \to [48, 24, 12, 0]$

4th. $i \to 3$

$result[3] = pref[3] * suff[3] \Rightarrow 8$

result → [48, 24, 12, 8] ✓