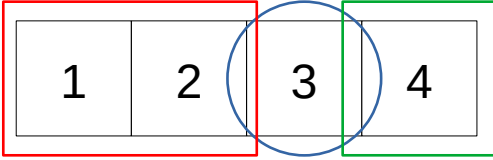


LeetCode #238

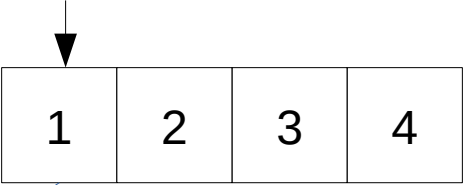
Product of Array Except Self

# Prefix and Postfix sum

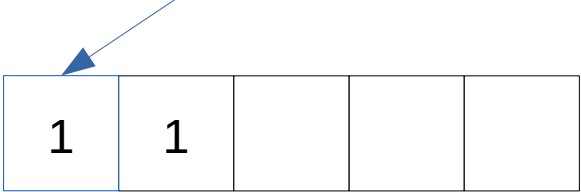


To get the product of the number 3 at index 2 excluding itself, we need to multiply the product of all elements before index 2 with the product of all elements after index 2. This can be achieved by computing the prefix product for indices 0 and 1, and the postfix product for index 3.

nums

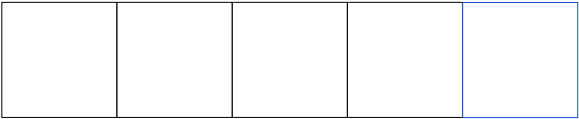


prefix



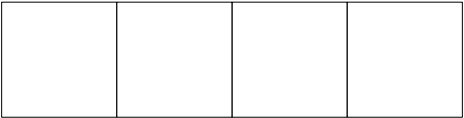
emp 0 1 2 3

postfix

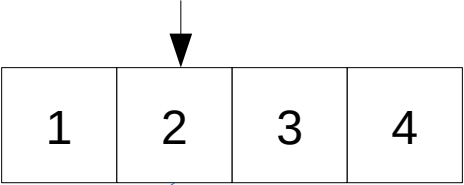


0 1 2 3 emp

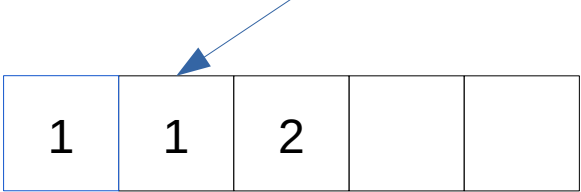
output



nums

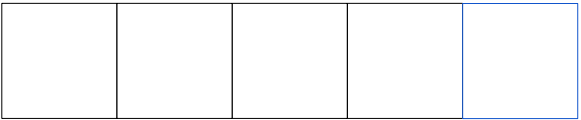


prefix



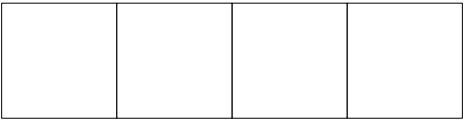
emp 0 1 2 3

postfix



0 1 2 3 emp

output



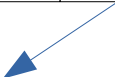
nums

1	2	3	4
---	---	---	---



prefix

1	1	2	6	
---	---	---	---	--



emp 0 1 2 3

postfix

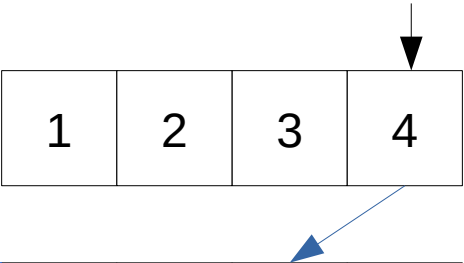
--	--	--	--	--

0 1 2 3 emp

output

--	--	--	--

nums



1	2	3	4
---	---	---	---

prefix

1	1	2	6	24
---	---	---	---	----

emp

0 1 2 3

postfix

--	--	--	--	--

0 1 2 3 emp

output

--	--	--	--

nums

1	2	3	4
---	---	---	---

prefix

1	1	2	6	24
---	---	---	---	----

emp

0 1 2 3

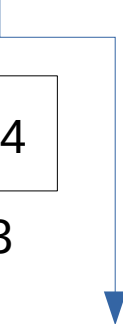
postfix

			4	1
--	--	--	---	---

0 1 2 3 emp

output

--	--	--	--



nums

1	2	3	4
---	---	---	---

prefix

1	1	2	6	24
---	---	---	---	----

emp

0 1 2 3

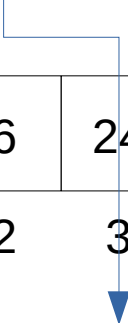
postfix

		12	4	1
--	--	----	---	---

0 1 2 3 emp

output

--	--	--	--





nums

1	2	3	4
---	---	---	---

prefix

1	1	2	6	24
---	---	---	---	----

emp

0 1 2 3

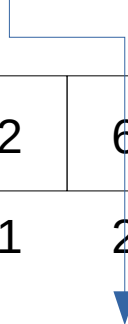
postfix

	24	12	4	1
--	----	----	---	---

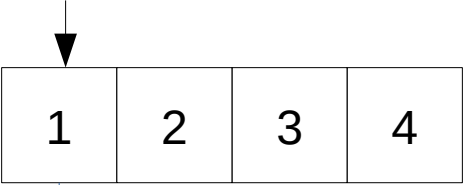
0 1 2 3 emp

output

--	--	--	--



nums



1	2	3	4
---	---	---	---

prefix

1	1	2	6	24
---	---	---	---	----

emp

0 1 2 3

postfix

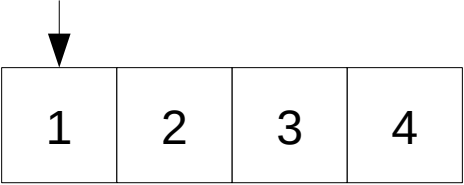
24	24	12	4	1
----	----	----	---	---

0 1 2 3 emp

output

--	--	--	--

nums



1	2	3	4
---	---	---	---

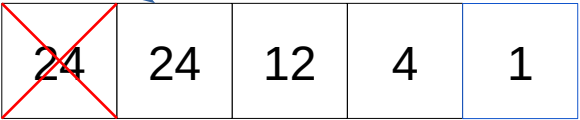
prefix

1	<del>1</del>	2	6	24
---	--------------	---	---	----

emp

0 1 2 3

postfix




<del>24</del>	24	12	4	1
---------------	----	----	---	---

0 1 2 3 emp

output

24			
----	--	--	--

nums



1	2	3	4
---	---	---	---

prefix

1	1	<del>2</del>	6	24
---	---	--------------	---	----

emp

0 1 2 3

postfix


24	<del>24</del>	12	4	1
----	---------------	----	---	---

0 1 2 3 emp

output

24	12		
----	----	--	--

nums



1	2	3	4
---	---	---	---

prefix

1	1	2	<del>6</del>	24
---	---	---	--------------	----

emp

0 1 2 3

postfix

24	24	<del>12</del>	4	1
----	----	---------------	---	---

0 1 2 3 emp

output

24	12	8	
----	----	---	--

nums

1	2	3	4
---	---	---	---

prefix

1	1	2	6	<del>24</del>
---	---	---	---	---------------

emp

0 1 2 3

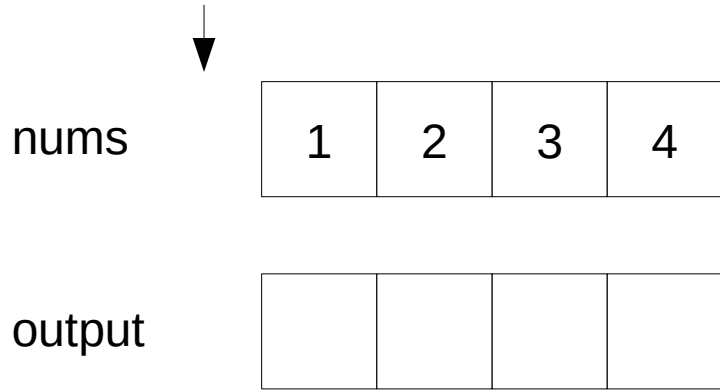
postfix

24	24	12	<del>4</del>	1
----	----	----	--------------	---

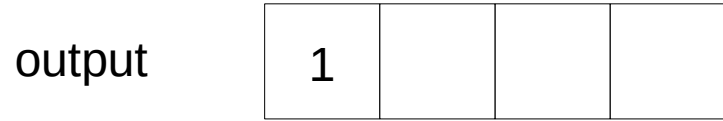
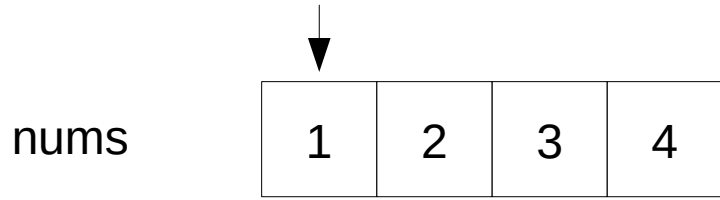
0 1 2 3 emp

output

24	12	8	6
----	----	---	---



It's possible to reduce extra memory usage for the prefix and postfix arrays by calculating these values on the fly.

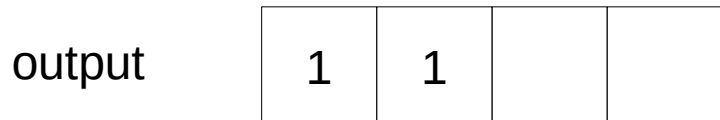
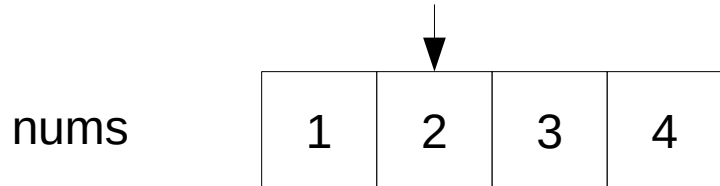


prefix = 1

output[i] = prefix = 1

prefix \*= nums[i] = 1 \* 1 = 1

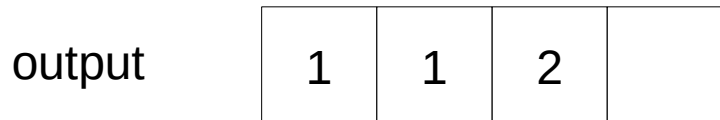
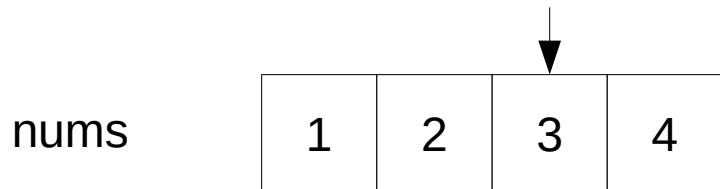




prefix = 1

output[i] = prefix = 1

prefix \*= nums[i] = 1 \* 2 = 2




prefix = 2

$\text{output}[i] = \text{prefix} = 2$

$\text{prefix} *= \text{nums}[i] = 2 * 3 = 6$

nums



1	2	3	4
---	---	---	---

output


1	1	2	6
---	---	---	---

prefix = 6

$\text{output}[i] = \text{prefix} = 6$

$\text{prefix} *= \text{nums}[i] = 6 * 4 = 24$

nums



1	2	3	4
---	---	---	---

output

1	1	2	6
---	---	---	---

postfix = 1

$\text{output}[i] *= \text{postfix} = 6 * 1 = 6$

$\text{postfix} *= \text{nums}[i] = 1 * 4 = 4$

nums



1	2	3	4
---	---	---	---

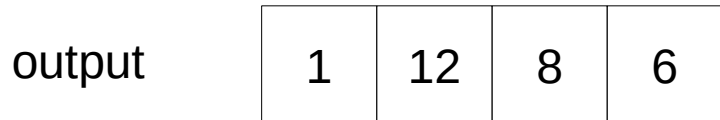
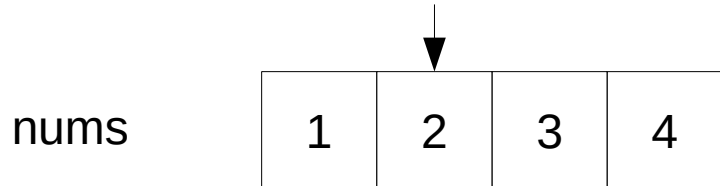
output

1	1	8	6
---	---	---	---

postfix = 4

$\text{output}[i] \ast \text{postfix} = 2 \ast 4 = 8$

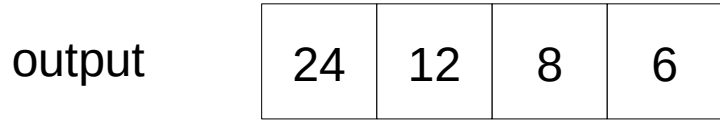
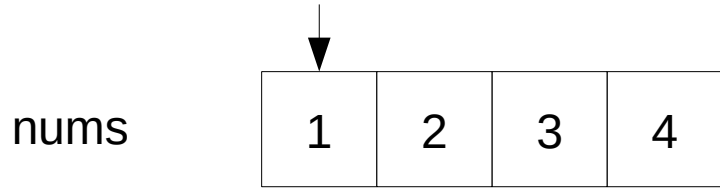
$\text{postfix} \ast \text{nums}[i] = 4 \ast 3 = 12$



postfix = 12

$\text{output}[i] *= \text{postfix} = 1 * 12 = 12$

$\text{postfix} *= \text{nums}[i] = 12 * 2 = 24$



postfix = 24

$\text{output}[i] *= \text{postfix} = 1 * 24 = 24$

$\text{postfix} *= \text{nums}[i] = 24 * 1 = 24$