Product of Array Except Self

Solution

Given an array `nums` of *n* integers where *n* > 1, return an array `output` such that `output[i]` is equal to the product of all the elements of `nums` except `nums[i]`.

Example:

```
Input:  [1,2,3,4]
Output: [24,12,8,6]
```

Constraint: It's guaranteed that the product of the elements of any prefix or suffix of the array (including the whole array) fits in a 32 bit integer.

Note: Please solve it **without division** and in O(*n*).

Follow up:

Could you solve it with constant space complexity? (The output array **does not** count as extra space for the purpose of space complexity analysis.)

Java

```java
class Solution {
    public int[] productExceptSelf(int[] nums) {
        int n = nums.length;
        int[] forward = new int[n];
        int[] backward = new int[n];

        forward[0] = 1;
        for(int i = 1; i < n; i++) {
            forward[i] = nums[i-1] * forward[i-1];
        }

        int temp = 1;
        for(int i = n-2; i >= 0; i--) {
            temp = nums[i+1] * temp;
            forward[i] *= temp;
        }

        return forward;
    }
}
```

Custom Testcase (Contribute)

Run Code Submit