75 Days of Code

Day10 Problem no: 283

Companies
Given an integer array nums, move all 0's to the end of it while maintaining the relative order of the non-zero elements.

Note that you must do this in-place without making a copy of the array.

Example 1:

Input: nums = [0,1,0,3,12]
Output: [1,3,12,0,0]

Example 2:

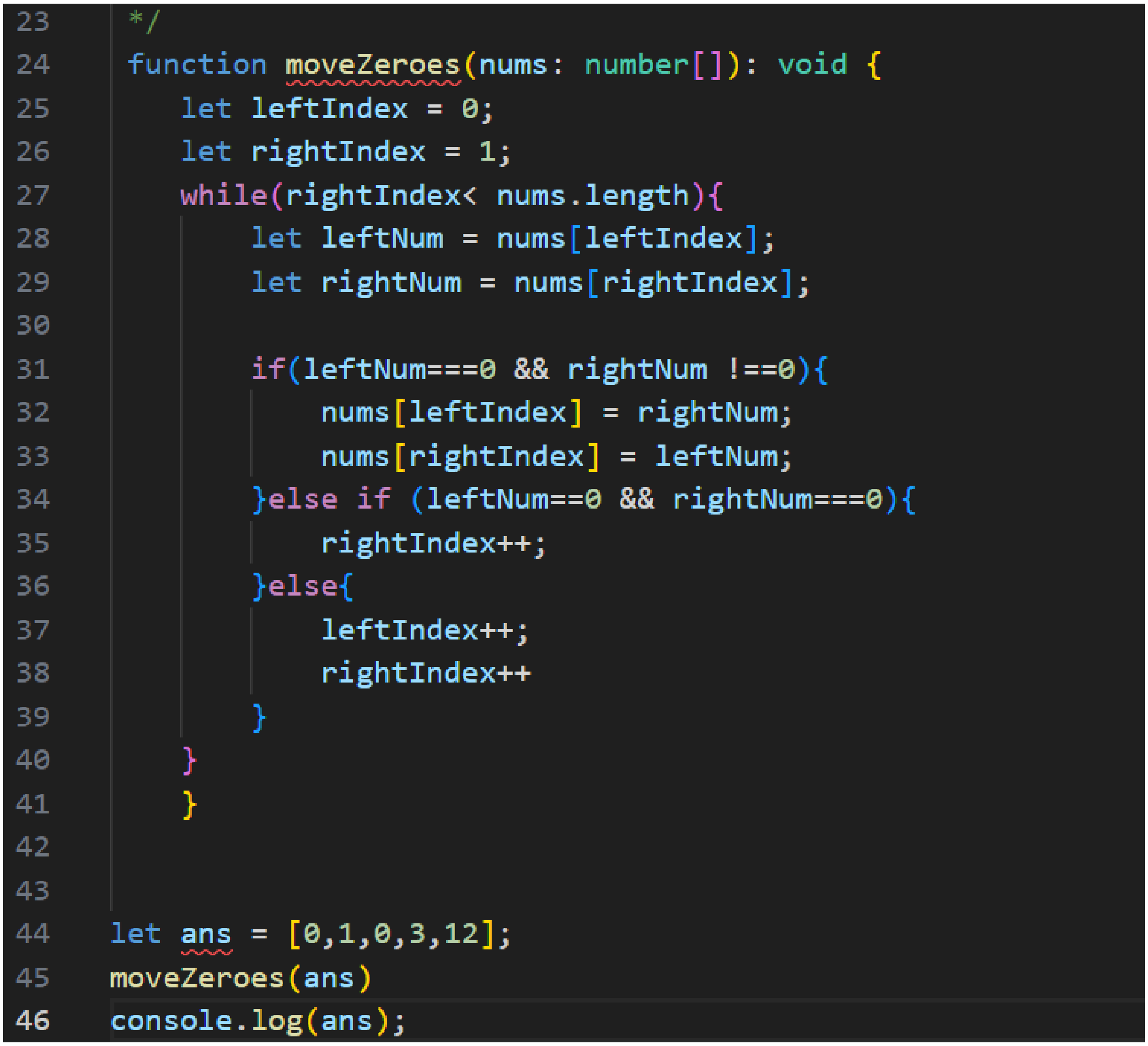
Input: nums = [0]
Output: [0]

For this problem ,

> **Note - we have to do it in place means have to modify original array instead of creating of new array**

1. Taking two pointers as left =0 , right = 1 and loop the array while right < array's length .
2. Swap the two element when first element is zero and second is not .
3. If both the pointers number is zero than increase right index
4. Else increase both the pointers

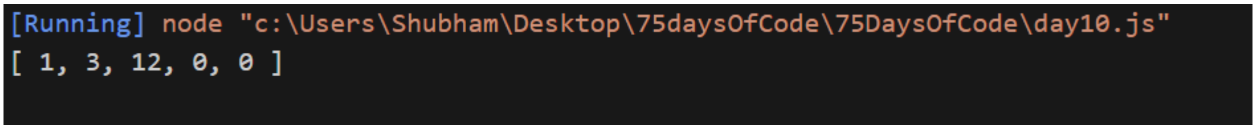
Shubham Agrahari



```
 */
 function moveZeroes(nums: number[]): void {
    let leftIndex = 0;
    let rightIndex = 1;
    while(rightIndex< nums.length){
        let leftNum = nums[leftIndex];
        let rightNum = nums[rightIndex];

        if(leftNum===0 && rightNum !==0){
            nums[leftIndex] = rightNum;
            nums[rightIndex] = leftNum;
        }else if (leftNum==0 && rightNum===0){
            rightIndex++;
        }else{
            leftIndex++;
            rightIndex++
        }
    }
    }


let ans = [0,1,0,3,12];
moveZeroes(ans)
console.log(ans);
```



```
[Running] node "c:\Users\Shubham\Desktop\75daysOfCode\75DaysOfCode\day10.js"
[ 1, 3, 12, 0, 0 ]
```