Move zeroes

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

Note: you must do this in-place without making copy of the array

Ex:

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

output: [1,3,12,0,0]

constraints:

→ $1 <= nums.length <= 10^4$

→ $-2^{31} <= nums[i] <= 2^{31}-1$

Algorithm:

1. Initilize j=0 → keeps track of next position to place a non-zero element

2. Iterate i from 0 to n-1:
   - → If nums[i] != 0 → move non-zero element to nums[j] and increment j
   - → This ensures all non-zero elements are shifted to front in order

3) After processing all elements, fill remaining positions from j to n-1 with 0's

Time complexity : O(n)

space complexity : O(1)

code:

```
int j=0                          //position for next non-zero element
for(int i=0 ; i<nums.length ; i++){
    if (nums[i] != 0){
        nums[j]=nums[i];        //place non-zero at index j
                                //move to next position
        j++;
    }
} //fill rest of array with zeroes
for (int i=j ; i<nums.length ; i++){
    nums[i]=0;
}
```