

### \* Q7. Remove elements :-

nums = [0, 1, 2, 2, 3, 0, 4, 2]      val = 2

- For the given array, we need to remove all occurrences of val from the array.
- Change array such that the first k elements do not contain val
- Return k.
- Do it in place

### \* Two pointer approach :-

nums = [0, 1, 2, 2, 3, 0, 4, 2]      val = 2

- Since we want to change the array in place, we can push the val to the very end of the array.
- Let's say the swapping task was done k times, then it means that first k elements do not contain val.

```
function solution(arr, value) {  
  let i = 0, j = arr.length - 1, k = 0;  
  while (i <= j) {  
    if (arr[i] == value) {  
      [arr[i], arr[j]] = [arr[j], arr[i]];  
      j--;  
    } else {  
      i++; k++;  
    }  
  }  
  return k;  
}
```

Time complexity :-  $O(n)$

Time complexity :-  $O(N)$   
Space complexity :-  $O(1)$