169. Majority Element

The majority element is the element that appears more than `⌊n / 2⌋` times. You may assume that the majority element always exists in the array.

T : O(n)

S : ~~O(n)~~ | O(1)

↳ hashmap

arr : 2 1 2 3 5

$\frac{n}{2}$ > freq

- 0 elements (no majority ele)
- 1 element (majority ele)

Moore's voting algo :

arr : 1 2 1 3 1 2 2 5 2 2 0 2 2 10 11 12

0 1 2 3 4 5 6 7 8 9 10 11 12

i

```
for(int i = 1; i < nums.length;i++) {
    if(nums[i] == val) {
        count++;

    }
    else {
        count--;
    }

    if(count == 0) {
        val = nums[i];
        count = 1;
    }
}
```

val = 1 2 1 3 1 2 2

count = 1 0 1 0 1 0 1 0 1 0 1 2 1 2 2 1 3

arr : 1 1 2 3 1 4 1 1

i

val = ~~1~~ ~~3~~ ~~1~~ ~~4~~ 1

count = ~~1~~ ~~1~~ 0 ~~1~~ 0 ~~1~~ 0 ~~1~~ 0 ~~1~~ 2

```
if (arr[i] == val) {
    count++;
}
else {
    count--;
}
if (count == 0) {
    val = arr[i];
    count = 1;
}
```

```
if (arr[i] == val) {
    count++;
}
else {
    count--;
}
if (count == 0) {
    val = arr[i];
    count = 1;
}
```

1 2 1 1 4

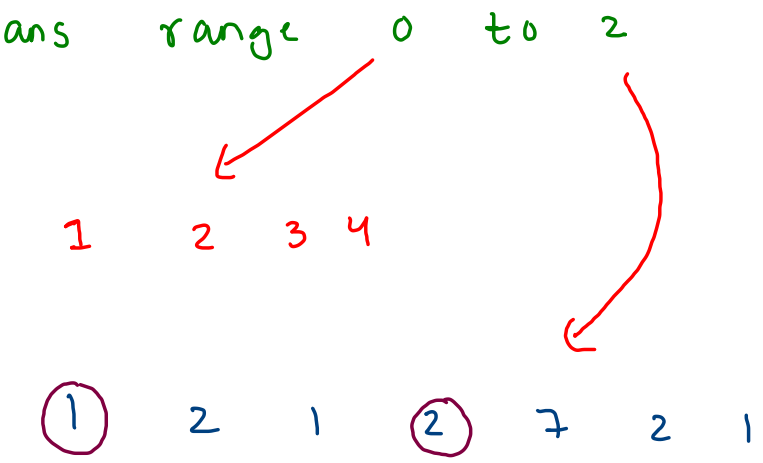
0 1 2 3 4

val = ~~x~~ ~~x~~ 1

count = ~~x~~ ~~x~~ 0 1 ~~x~~ 1

229. Majority Element II

Given an integer array of size `n`, find all elements that appear more than `⌊ n/3 ⌋` times.



1	2	7	1	1	2	3
0	1	2	3	4	5	6

i

v1 = 1

c1 = ~~2~~ 1

v2 = 2

c2 = ~~1~~ 0

```
if (arr[i] == v1) {
    c1++;
}
else if(arr[i] == v2) {
    c2++;
}
else {
    if (c1 == 0) -> v1 = arr[i], c1 = 1;
    else if (c2 == 0) -> v2 = arr[i], c2 = 1;
    else {
        c1--; c2--;
    }
}
```

Majority Element General

Easy

Prev Next

1. Given an array of size 'N' and an element K.
2. Task is to find all elements that appears more than N/K times in array.
3. Return these elements in an ArrayList in sorted order.

```
8
3 1 2 2 1 2 3 3
4
```

3 - 3

1 - 2

2 - 3

$\frac{n}{k} = \frac{8}{4} = 2$

[2, 3]

345. Reverse Vowels of a String

Easy 1367 1706 Add to List Share

Given a string `s`, reverse only all the vowels in the string and return it.

The vowels are `'a'`, `'e'`, `'i'`, `'o'`, and `'u'`, and they can appear in both cases.

O(n) : T

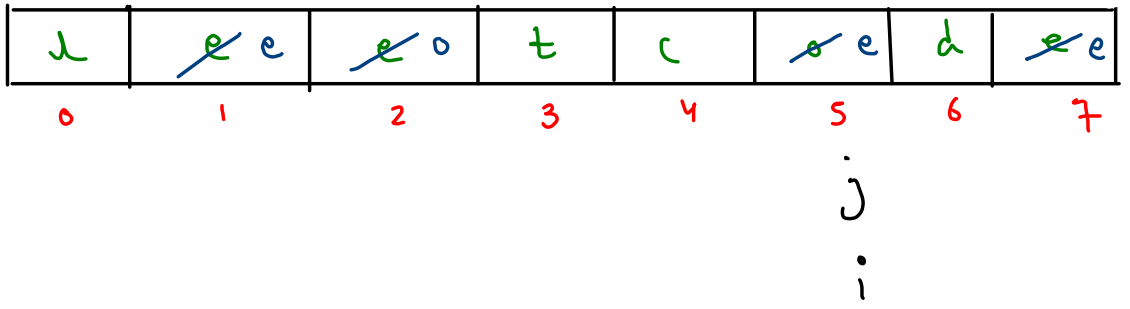
a, e, i, o, u

A, E, I, O, U

l e e t c o d e

ans: l e o t c e d e

l e e t c o d e



Complex Number Multiplication

$$c_1 = a + bi$$

$$c_2 = c + di$$

$$c_1 * c_2 = (a + bi) * (c + di)$$

$$= (ac - bd) + i(ad + bc)$$