Majority Element -02

A majority Element is having a $\left(\text{frequency} > \frac{N}{3}\right)$ (N is size of array)

* We use same approach - "Boyer's moore voting algo"

(But with additional Components)

* previously if two distinct elements occured, we cancelled them
* Now, if three distinct elements occurs, we will cancel them

Let's split the solution in 2 phases

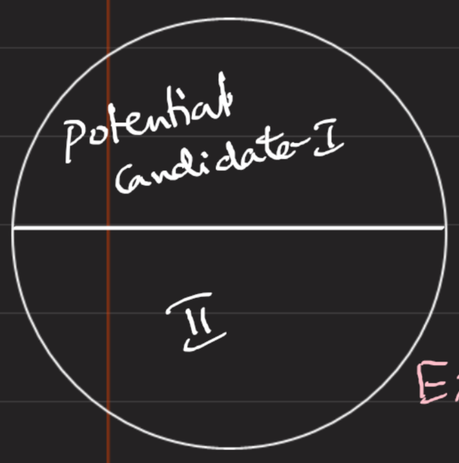
① Identifying potential candidates
② Verifying the candidates

① Identifying potential candidates

* Use 2 variables to store potential Candidates & their counts
* Traverse the array
  ① If one counter is zero, update it with current element and set freq to 1
  ② If current element matches with one of the candidates increment its count
  ③ Otherwise, decrement both candidates counts

② Verifying the candidates

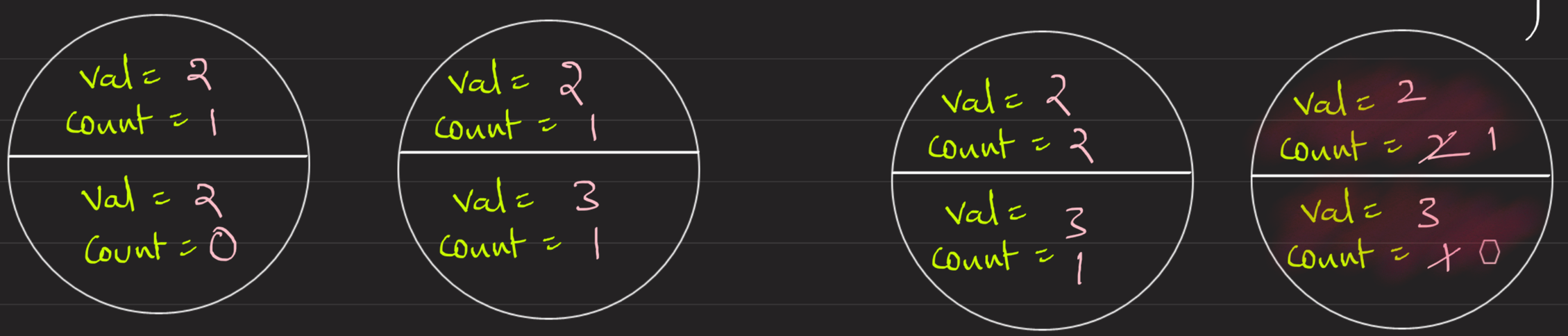
* Initialize count for the assumed two potential candidates
* Traverse array to count every candidate occurance
* Check if any of those 2 candidates count is $> \frac{n}{3}$



Ex: arr[11] = (2, 3, 2, 4, 2, 3, 5, 3, 3, 2, 4)

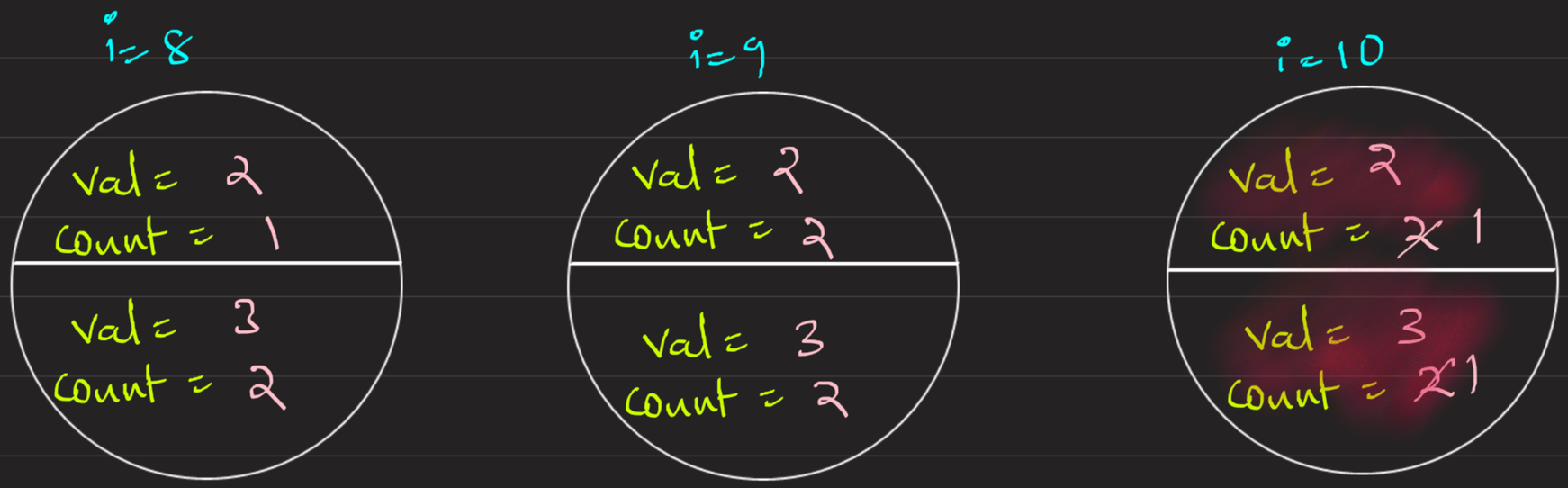
0 1 2 3 4 5 6 7 8 9 10

if we observe 3 candidates are distinct, we reduce count



Ex: arr[11] = (2, 3, 2, 4, 2, 3, 5, 3, 3, 2, 4)

0 1 2 3 4 5 6 7 8 9 10



* At last, for value = 2, 3 ⇒ we are having counts > 0

So, these are possible candidates for majority elements

Hence, traverse through the array, and check whether {2, 3} are appearing ($> \frac{n}{3}$) times or not

* Which ever element from {2, 3} appears ($> \frac{n}{3}$) times that element is a majority element

```
for(int i = 1;i<n;i++){
    if(nums[i] == val1)cnt1++;
    else if(nums[i] == val2)cnt2++;
    else if(cnt1 == 0){
        val1 = nums[i];
        cnt1++;
    }
    else if(cnt2 == 0){
        val2 = nums[i];
        cnt2++;
    }
    else if(val1 != nums[i] && val2 != nums[i]){
        cnt1--;
        cnt2--;
    }
}

int candidate1 = 0;
int candidate2 = 0;

for(int i = 0;i<n;i++){
    if(val1 == nums[i])candidate1++;
    else if(val2 == nums[i])candidate2++;
}
```

→ phase-I (as mentioned starting)

→ phase-II

↓

Check which candidate is $> \frac{N}{3}$ & add it to your answer