Day-11

Python DSA

Leetcode 229 Majority Element

https://leetcode.com/problems/majority-element-ii/description/

Bruteforce

```
from typing import List

class Solution:

    def majorityElement(self, nums: List[int]) -> List[int]:

        result = []

        n = len(nums)

        for i in range(n):

            if len(result) == 0 or result[0] != nums[i]:

                count = 0

                for j in range(n):

                    if nums[j] == nums[i]:

                        count += 1

                if count > n // 3:

                    result.append(nums[i])

            if len(result) == 2:

                break

        return result


sol = Solution()

nums = [3, 2, 3]
```

```
print(sol.majorityElement(nums))
```

TC- O(N^2)

SC- O(1)

Better

```
def majorityElement(nums: List[int]) -> List[int]:
    n= len(nums) hash_map = dict()
    for num in nums:
        hash_map[num]= hash_map.get(num,0)+1
    result=[]
    for key in hash_map:
        if hash_map[key] > n//3:
            result.append(key)
    return result
nums = [3, 2, 3]
print(majorityElement(nums))
```

TC- O(N)

SC- O(N)

Optimal Solution

```
class Solution:
    def majorityElement(self, nums: List[int]) -> List[int]:
        n= len(nums)
        count1=0
        count2=0
        ele1= float("-inf")
        ele2= float("-inf")
        for i in range(n):
            if count1==0 and nums[i] != ele2:
                count1=1
                ele1= nums[i]
            elif count2==0 and nums[i] != ele1:
                count2=1
                ele2 = nums[i]
            elif nums[i] == ele1:
                count1+=1
            elif nums[i]==ele2:
                count2 +=1
            else:
                count1 -=1
                count2 -=1
        count1=0
        count2=0
        for i in range(n):
            if nums[i]==ele1:
                count1 +=1
            elif nums[i] == ele2:
                count2+=1
        result=[]
        if count1 > n//3:
            result.append(ele1)
        if count2 > n//3:
            result.append(ele2)
        return result
```

TC- O(N)

SC-O(1)