771. Jewels and Stones

Jewels and Stones

```
class Solution:
    def numJewelsInStones(self, J: str, S: str) -> int:
        return sum(s in J for s in S)
```

961. N-Repeated Element in Size 2N Array

N-Repeated Element in Size 2N Array

```
class Solution:
    def repeatedNTimes(self, A: List[int]) -> int:

        for i in range(len(A)):
            s=0
            for j in range(len(A)):
                if A[i]==A[j]:
                    s+=1

            if s==(len(A)/2):
                return A[i]
```

1207. Unique Number of Occurrences

Unique Number of Occurrences

```
class Solution:
    def uniqueOccurrences(self, arr: List[int]) -> bool:
        a=Counter(arr)
        b=set(a.values())

        c=len(b)
        d=len(a)

        if c==d:
            return True
```

1365. How Many Numbers Are Smaller Than the Current Number

How Many Numbers Are Smaller Than the Current Number

```
class Solution:
    def smallerNumbersThanCurrent(self, nums: List[int]) -> List[int]:
        ans=[]
        for i in range(len(nums)):
            count=0

            for num in nums:
                if nums[i]>num:
                    count+=1

            ans.append(count)
        return ans
```

1512. Number of Good Pairs

Number of Good Pairs

```
class Solution:
    def numIdenticalPairs(self, nums: List[int]) -> int:
        n=len(nums)
        count={}
        ans=0

        for i in nums:
            if i not in count:
                count[i]=1

            else:
                ans+=count[i]
                count[i]+=1

        return ans
```