1) class Solution:

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

s=set(nums)

for i in s:

if nums.count(i)==1:

return i

else:

continue

2) class Solution:

def sortByBits(self, arr: List[int]) -> List[int]:

arr.sort()

k=[]

for i in arr:

a=bin(i).replace("0b", "")

l=str(a)

k.append([l.count('1'),arr.index(i)])

k.sort()

array=[]

for i in range(len(k)):

a=k[i]

b=a[1]

array.append(arr[b])

return array

3) class Solution:

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

l=[]

s=set(nums)

for i in s:

if nums.count(i)==1:

l.append(i)

return l

4) from itertools import combinations

class Solution:

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

l=[]

for i in range(len(nums)+1):

l+= [list(j) for j in combinations(nums, i)]

return l