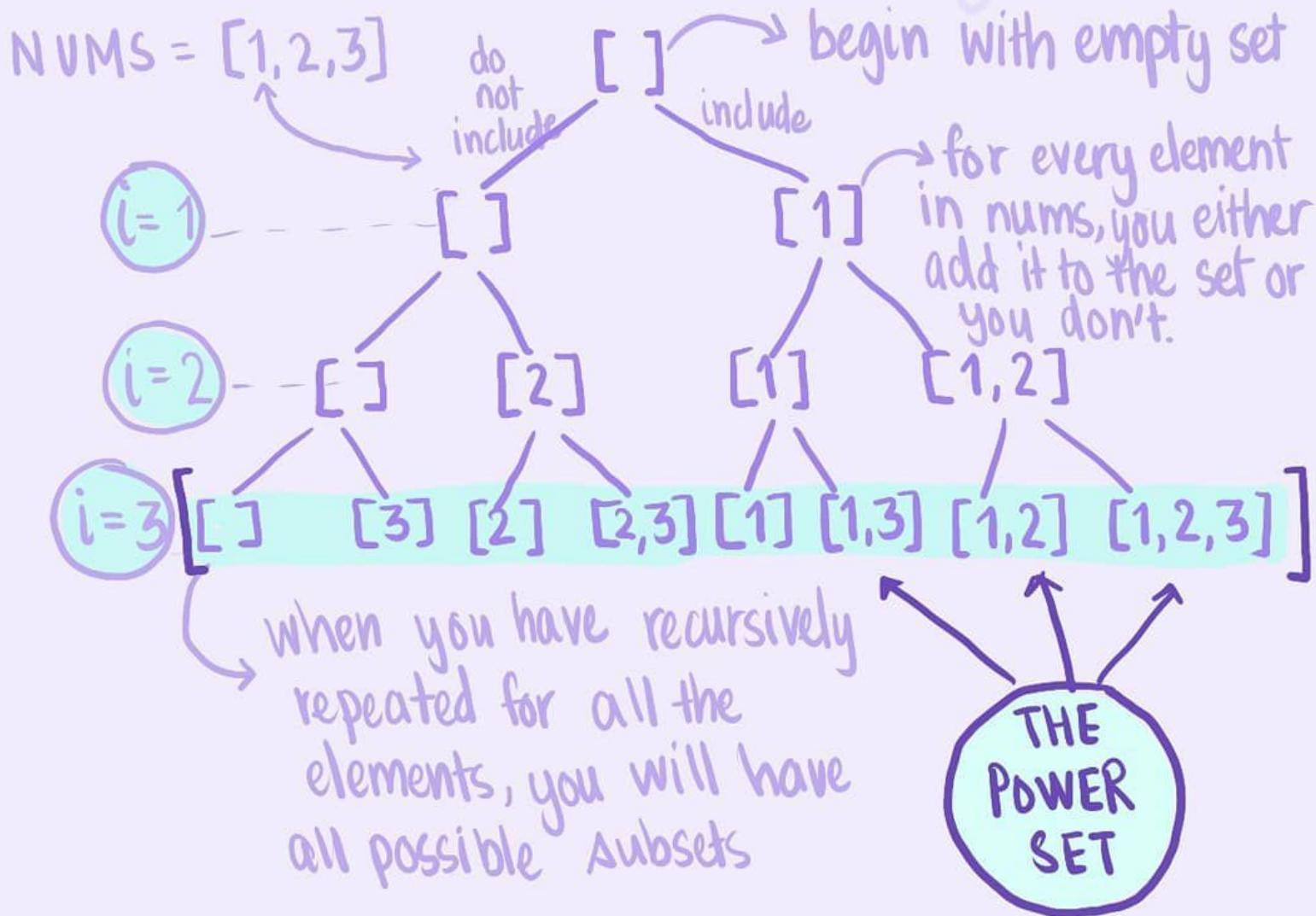


LEETCODE

78. SUBSETS

Given a set of distinct integers, `nums`
return all possible subsets (the power set)

BACKTRACKING



The **CODE** Dose

```
def subsets_util(nums, i, s, powerSet):  
    if i >= len(nums):  
        powerSet.append(s)  
    else:  
        temp = s.copy()  
        temp.append(nums[i])  
        subsets_util(nums, i + 1, temp, powerSet)  
        subsets_util(nums, i + 1, s, powerSet)  
  
class Solution:  
    def subsets(self, nums: List[int]) -> List[List[int]]:  
        powerSet = []  
        if len(nums) == 0:  
            return []  
        subsets_util(nums, 0, [], powerSet)  
        return powerSet
```

iterate through nums

add set to powerSet if reached end of nums

include

move to next element

not include

recursive function

begin with empty set