

# 78. Subsets

## Code Walkthrough

```

1 def subsets(nums):
2     n = len(nums)
3     res, sol = [], []
4
5     def backtrack(i):
6         if i == n:
7             res.append(sol[:]) } Base case
8             return
9
10        backtrack(i+1) } Option 1: Don't Include
11
12        sol.append(nums[i]) } Option 2: Include
13        backtrack(i+1)
14        sol.pop()
15
16    backtrack(0)
17    return res

```

### Process

$\text{nums} = [1, 2]$

Recursive  
Call  
Stack



backtrack(0)  $i=0$  [ ]

Line 1b

$\text{sol} = []$   
 $\text{res} = []$



backtrack(0)  $i=0$  [ ]

Line 10  $i \neq n$ ; so we start our first recursion w/ option 1

backtrack(1)  $i=1$  [ ]

$\text{sol} = []$   
 $\text{res} = []$



backtrack(0)  $i=0$  [ ]

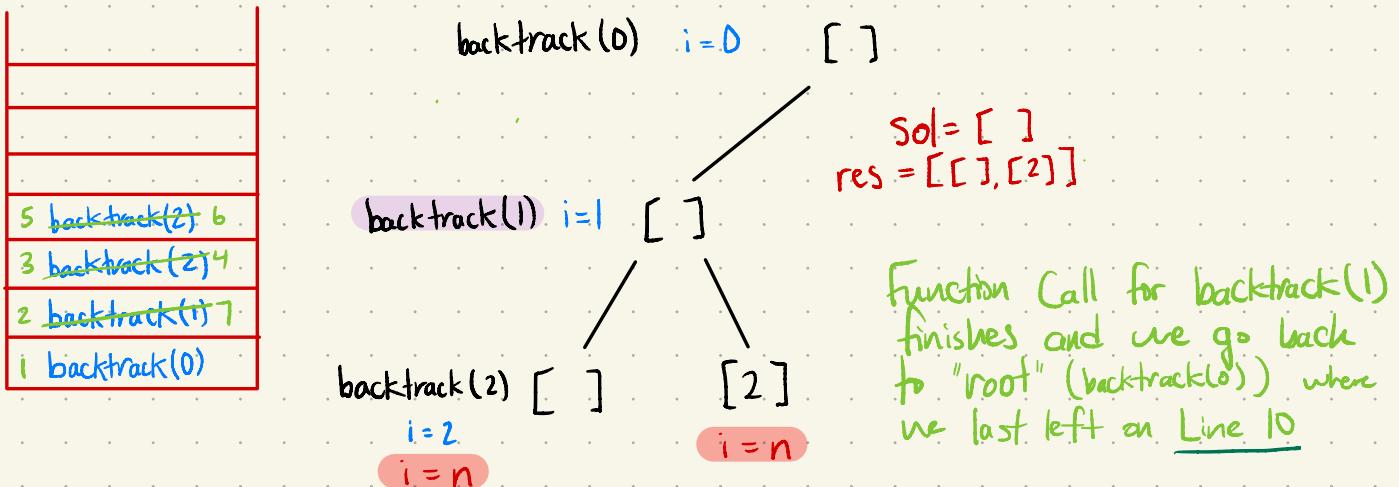
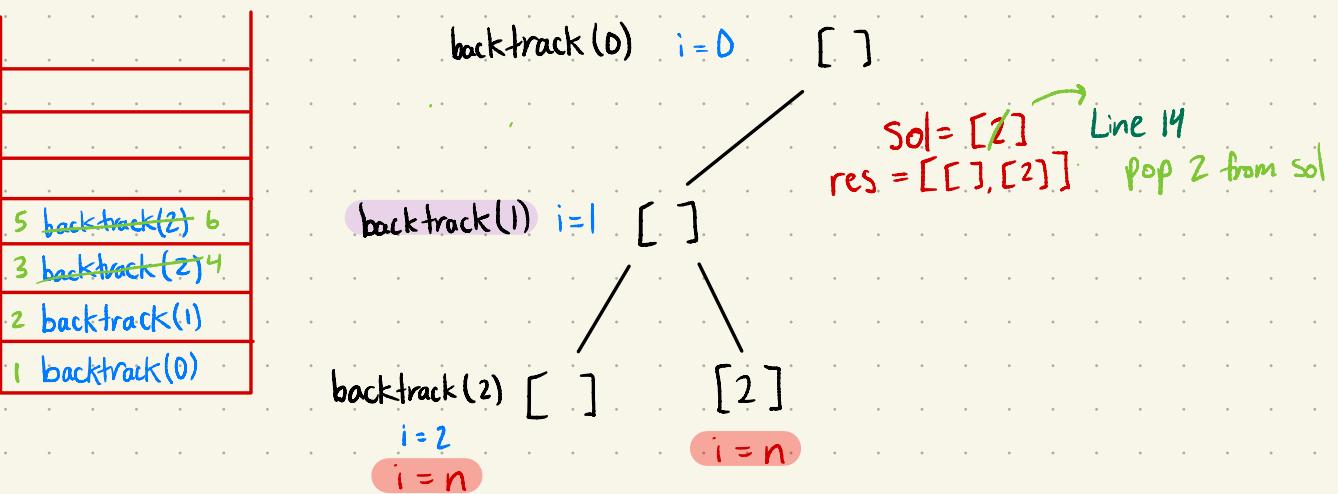
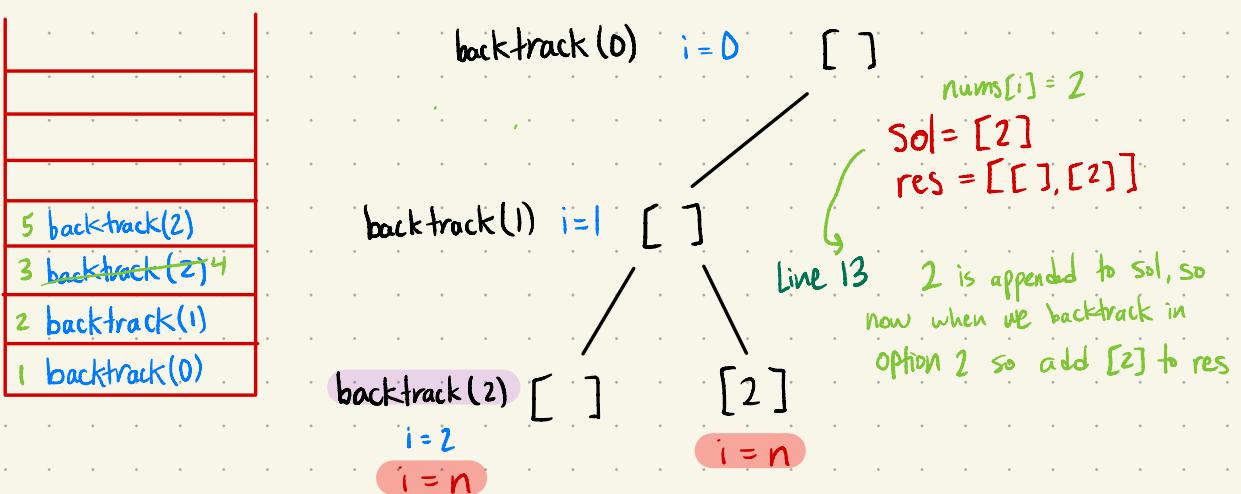
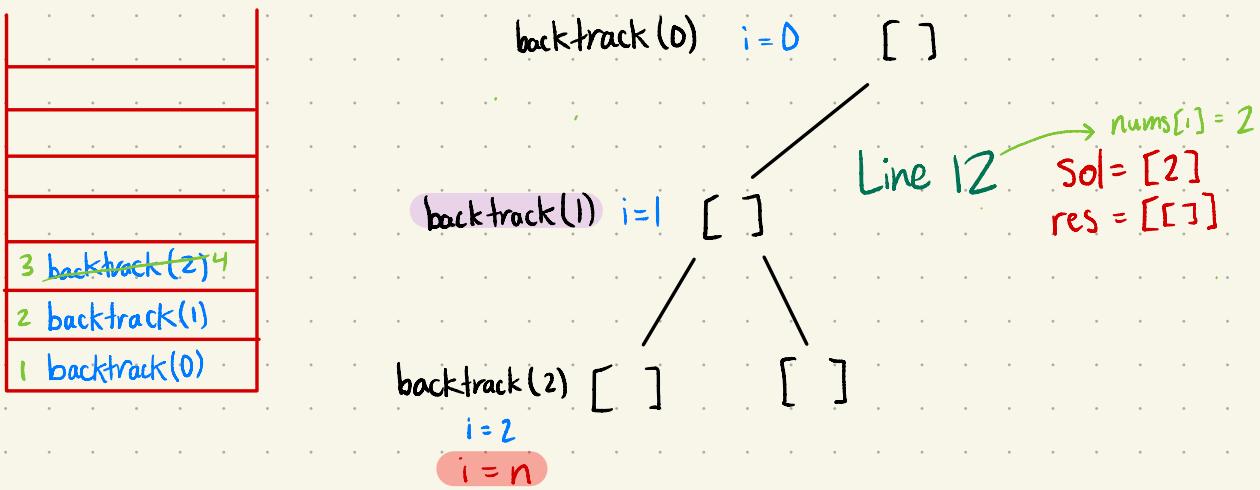
backtrack(1)  $i=1$  [ ]

$\text{sol} = []$   
 $\text{res} = [[]]$

backtrack(2) [ ]  $i=n$

Line 6-8

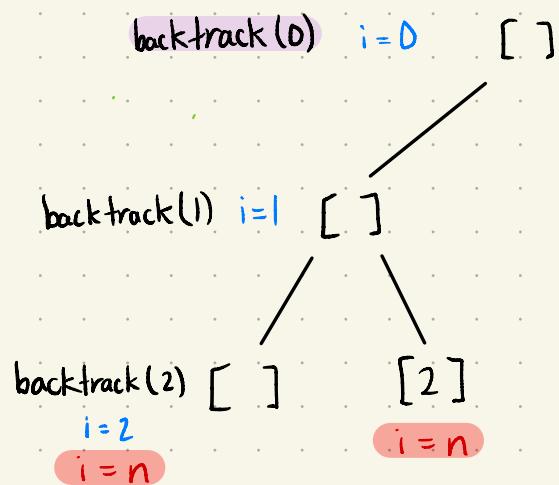
Since  $i=n$ , add the initial empty  $\text{sol}$  array to  $\text{res}$  then return to get out the function call



```

5 backtrack(2) 6
3 backtrack(2) 4
2 backtrack(1) 7
1 backtrack(0)

```

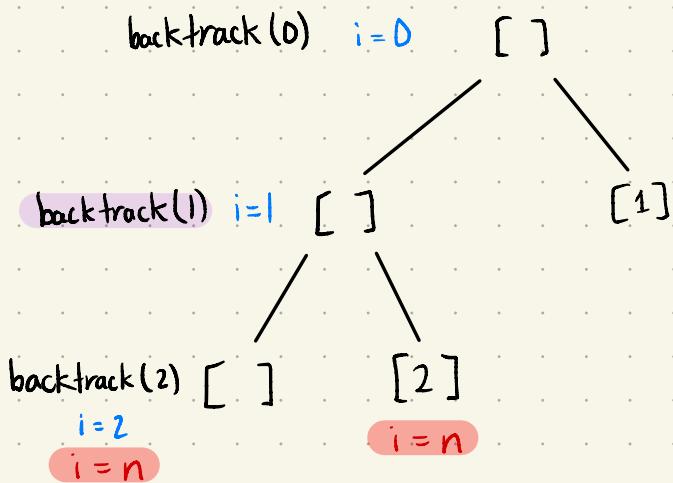


Line 12  $\text{nums}[i]=1$   
 $\text{Sol}=[1]$   
 $\text{res}=[[ ], [2]]$

```

8 backtrack(1)
5 backtrack(2) 6
3 backtrack(2) 4
2 backtrack(1) 7
1 backtrack(0)

```



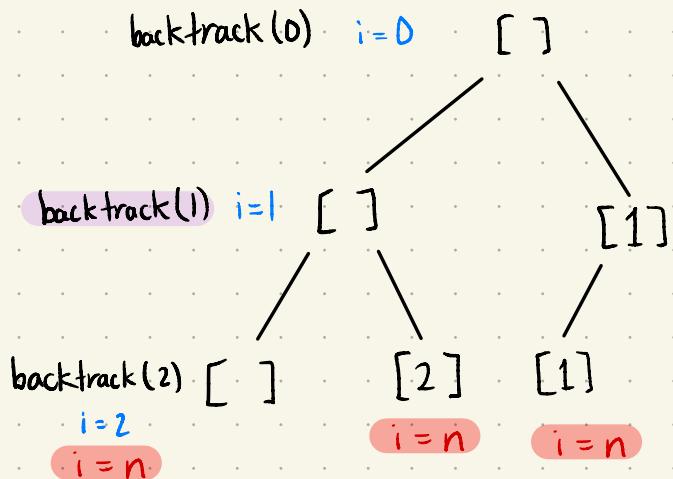
$\text{Sol}=[1]$   
 $\text{res}=[[ ], [2]]$

Line 13 start  
reversing the right side

```

9 backtrack(2) 10
8 backtrack(1)
5 backtrack(2) 6
3 backtrack(2) 4
2 backtrack(1) 7
1 backtrack(0)

```



Line b-8  
add [1] to res

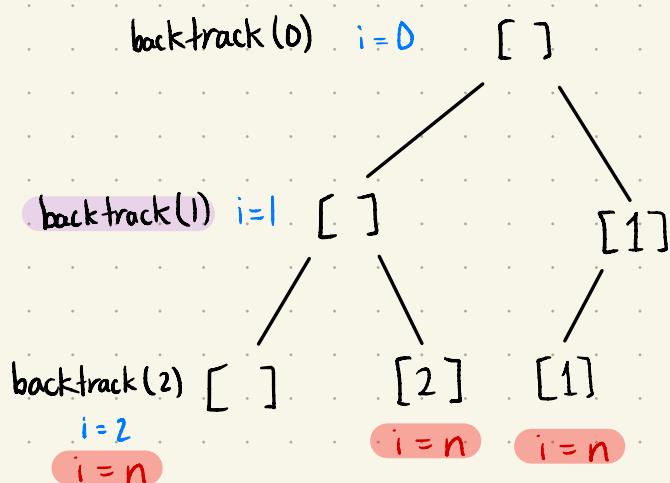
$\text{Sol}=[1]$   
 $\text{res}=[[ ], [2], [1]]$

and return function

```

9 backtrack(2) 10
8 backtrack(1)
5 backtrack(2) 6
3 backtrack(2) 4
2 backtrack(1) 7
1 backtrack(0)

```

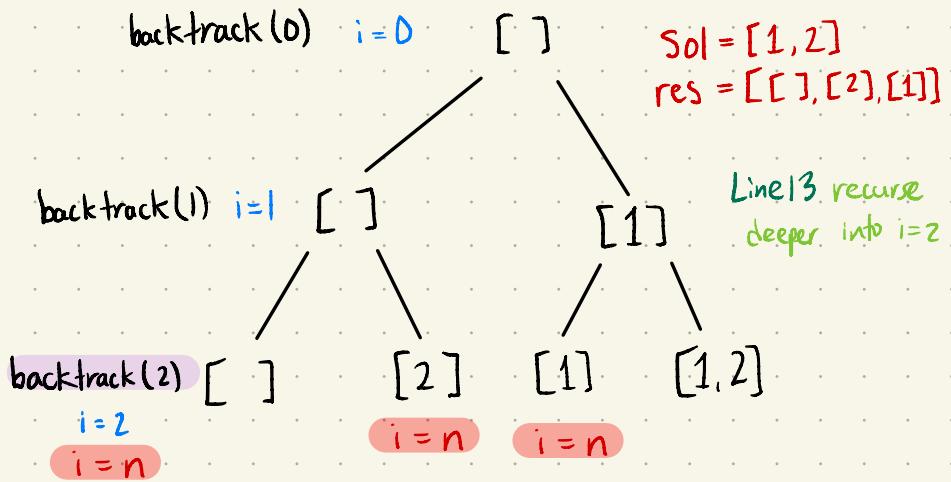


Line 12  $\text{nums}[i]=2$   
 $\text{Sol}=[1, 2]$   
 $\text{res}=[[ ], [2], [1]]$

```

11 backtrack(2)
9 backtrack(2) 10
8 backtrack(1)
5 backtrack(2) 6
3 backtrack(2) 4
2 backtrack(1) 7
1 backtrack(0)

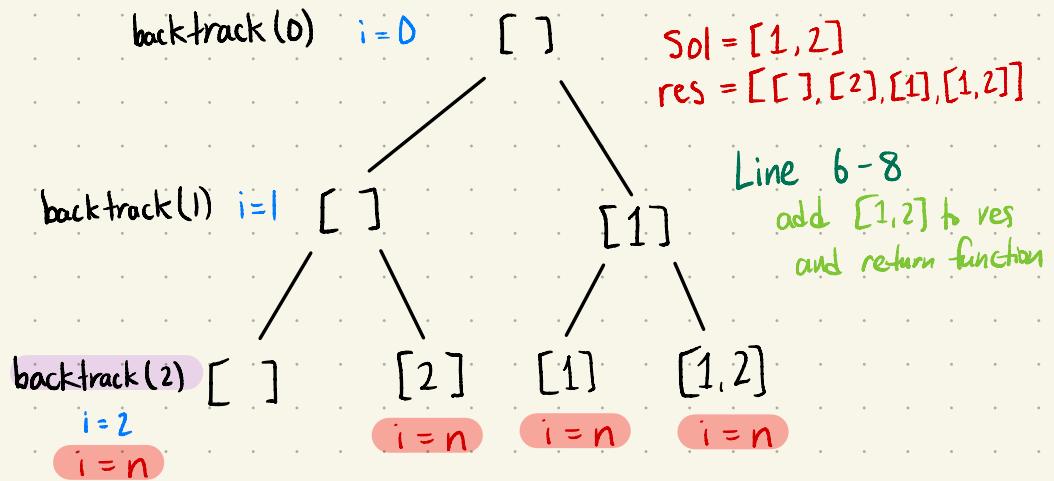
```



```

11 backtrack(2) 12
9 backtrack(2) 10
8 backtrack(1)
5 backtrack(2) 6
3 backtrack(2) 4
2 backtrack(1) 7
1 backtrack(0)

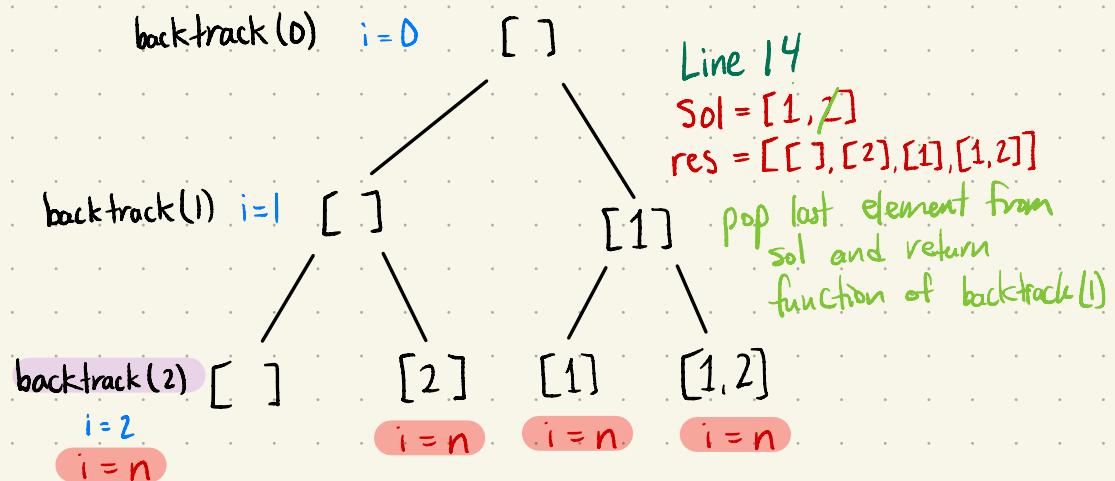
```



```

11 backtrack(2) 12
9 backtrack(2) 10
8 backtrack(1) 13
5 backtrack(2) 6
3 backtrack(2) 4
2 backtrack(1) 7
1 backtrack(0)

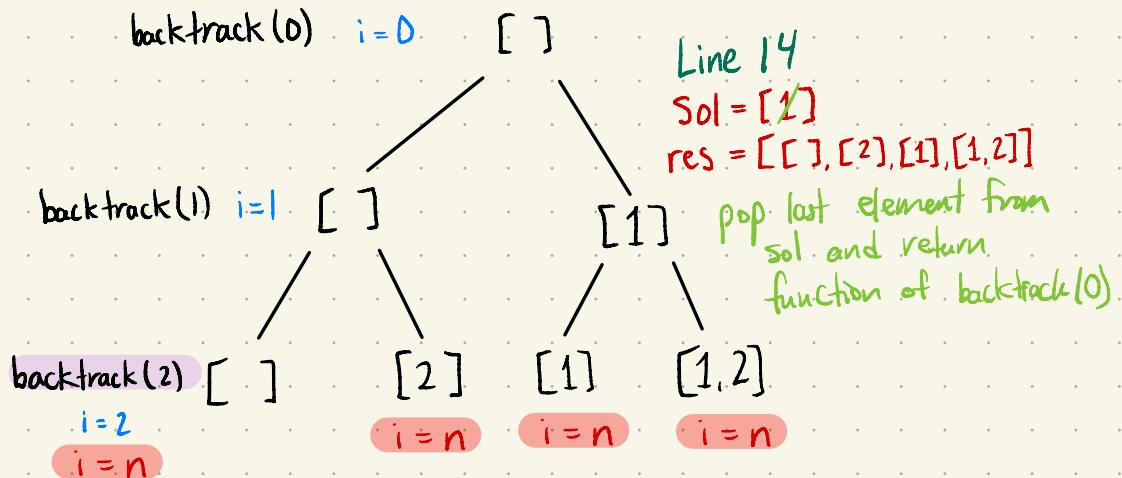
```



```

11 backtrack(2) 12
9 backtrack(2) 10
8 backtrack(1) 13
5 backtrack(2) 6
3 backtrack(2) 4
2 backtrack(1) 7
1 backtrack(0) 13

```



Call Stack is Now Empty!

Line 17 return res array

res = [[ ], [2], [1], [1, 2]]