**Justin\_Tree\_Depth-firstSearch\_0108. Convert Sorted Array to Binary Search Tree**

**Concept:**

先把 [左半部, root, 右半部] 分出來

用遞迴往每個 root.left、root.right 找

**Code:**

# Definition for a binary tree node.

# class TreeNode:

# def \_\_init\_\_(self, x):

# self.val = x

# self.left = None

# self.right = None

class Solution:

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

def split\_lr(list\_nums):

if not list\_nums:

return None

left = list\_nums[0:int(len(list\_nums)/2)]

right = list\_nums[int(len(list\_nums)/2)+1:]

middle = list\_nums[int(len(list\_nums)/2)]

root = TreeNode(middle)

root.left = split\_lr(left)

root.right = split\_lr(right)

return root

return split\_lr(nums)