26. You are given an integer array nums with no duplicates. A maximum binary tree can be built recursively from nums using the following algorithm: Create a root node whose value is the maximum value in nums. Recursively build the left subtree on the subarray prefix to the left of the maximum value. Recursively build the right subtree on the subarray suffix to the right of the maximum value. Return the maximum binary tree built from nums.

Program: # Define the TreeNode class to represent each node in the binary tree

```
class TreeNode:
    def _init_(self, val=0, left=None, right=None):
        self.val = val
        self.left = left
        self.right = right
def constructMaximumBinaryTree(nums):
    # Base case: if nums is empty, return None
    if not nums:
        return None

    max_index = nums.index(max(nums))
        root = TreeNode(nums[max_index])
        root.left = constructMaximumBinaryTree(nums[:max_index])
        root.right = constructMaximumBinaryTree(nums[max_index + 1:])
    return root
def printTree(root):
    if root:
        print(root.val, end=' ')
        printTree(root.left)
        printTree(root.right)
nums = [3, 2, 1, 6, 0, 5]
root = constructMaximumBinaryTree(nums)
printTree(root)
```

OUTPUT:-

```
6 3 2 1 5 0
=== Code Execution Successful ===
```

time complexity :O(n)