

4. Convert sorted list to binary search tree (LeetCode)

```
class Solution {  
public:  
    TreeNode* sortedListToBST(ListNode* head) {  
        if (!head) return nullptr;  
        return build(head, nullptr);  
    }  
  
    TreeNode* build(ListNode* start, ListNode* end) {  
        if (start == end) return nullptr;  
  
        ListNode* slow = start;  
        ListNode* fast = start;  
  
        while (fast != end && fast->next != end) {  
            slow = slow->next;  
            fast = fast->next->next;  
        }  
  
        TreeNode* root = new TreeNode(slow->val);  
        root->left = build(start, slow);  
        root->right = build(slow->next, end);  
  
        return root;  
    }  
};
```

OUTPUT:

```
✓ Testcase | >_ Test Result  
Case 1 Case 2 +  
  
head =  
[-10,-3,0,5,9]
```

```
✓ Testcase | >_ Test Result  
Case 1 Case 2 +  
  
head =  
[]
```