

哈希表

2022年8月18日 10:25

关于两数之和

在某数据结构中（可能是链表可能是数），找到两个元素，使其和为 val，
若存在返回true，不存在返回false

将当前值 cur->val 存入 hash 中，并搜索 hash 中是否有 val-cur->val 的存在，存在则返回 true，不存在则继续搜索，直到遍历完毕，则返回 false

653. 两数之和 IV - 输入 BST

难度 简单  422     

给定一个二叉搜索树 root 和一个目标结果 k，如果 BST 中存在两个元素且它们的和等于给定的目标结果，则返回 true。

此题中可使用哈希表

```
class Solution {
public:
    map<int, int> hash;
    bool findTarget(TreeNode* root, int k) {
        if(!root)
            return false;
        if(hash.count(k-root->val))
            return true;
        hash[root->val]=1;
        return findTarget(root->left,k)||findTarget(root->right,k);
    }
};
```

也可以考虑使用搜索二叉树的性质

例如中序遍历后是严格递增的序列，然后使用双指针即可