

给定二叉搜索树（BST）的根节点和一个值。你需要在BST中找到节点值等于给定值的节点。返回以该节点为根的子树。如果节点不存在，则返回 NULL。

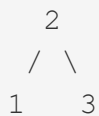
例如，

给定二叉搜索树：



和值：2

你应该返回如下子树：



在上述示例中，如果要找的值是 5，但因为没有节点值为 5，我们应该返回 NULL。

思路，进行搜索二叉树的遍历，遍历到相等节点后，返回该节点，否则，返回NULL

思路：

```
/**
 * Definition for a binary tree node.
 * struct TreeNode {
 *     int val;
 *     TreeNode *left;
 *     TreeNode *right;
 *     TreeNode(int x) : val(x), left(NULL), right(NULL) {}
 * };
 */

class Solution {
public:
    TreeNode* searchBST(TreeNode* root, int val) {
        if(root==NULL){
            return NULL;
        }
        if(root->val==val){
```

```
        return root;
    }else if(root->val<val){
        if(root->right!=NULL){
            return searchBST(root->right,val);
        }

    }else{
        if(root->left!=NULL){
            return searchBST(root->left,val);
        }
    }
    return NULL;
}
};
```