剑指07 重建二叉树

方法：分治，实现：递归，深度优先X序遍历

TODO：使用hash，再做一次。

```
// 2个小时乱写的，中间关于index的处理出了问题，调了很久bug。
// find操作可以用hash更优
class Solution {
public:
    TreeNode* dfs(int &index, vector<int>& preorder, vector<int>& inorder, pair<int, int> range) {
        if (range.first > range.second) return NULL;
        if (index == preorder.size()) return NULL;

        vector<int>::iterator it = find(inorder.begin() + range.first, inorder.begin() + range.second + 1, preorder[index]);
        TreeNode* node = new TreeNode(preorder[index]);
        int x = distance(inorder.begin(), it);
        node->left = dfs(++index, preorder, inorder, make_pair(range.first, x - 1));
        if(node->left == NULL) index--;
        node->right = dfs(++index, preorder, inorder, make_pair(x + 1, range.second));
        if(node->right == NULL) index--;
        return node;
    }

    TreeNode* buildTree(vector<int>& preorder, vector<int>& inorder) {
        if (preorder.empty()) return NULL;
        int index = 0;
        return dfs(index, preorder, inorder, make_pair(0, inorder.size() - 1));
    }
};

```