

数据结构代码

数据结构代码

树

判断二叉树是不是AVL

AVL中的第k个数

树

判断二叉树是不是AVL

```
int isBalanceTree(BinaryTreeNode<T>*root ,int &height){
    int left_height,right_height;
    if(root==NULL){height = 0;return 1}
    if(root->leftChild == NULL & root->rightChild == NULL){
        height = 1;
        return 1;
    }
    if(!isBalanceTree(root->leftChild))return 0;
    if(!isBalanceTree(root->rightChild))return 0;
    height = (left_height > right_height)? left_height+1:right_height;
    return math.abs(left_height - right_height)<2;
}
```

AVL中的第k个数

AVL树结点中增设一个域lsize，表示左子树的结点数+1，求第k个结点的位置

```
BinaryTreeNode<T>* search_k(BinaryTreeNode<T>*root, int k){
    if(root==NULL || k<1){return NULL;}
    if(root->lsize==k)return root;
    if(root->lsize > k )return search_k(root->leftChild,k);
    else return search_k(root->right,k-root->lsize);
}
```