



链滴

二叉树的递归遍历

作者: [ellenbboe](#)

原文链接: <https://ld246.com/article/1565226134610>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

二叉树遍历

前序遍历

```
static List<Integer> list = new ArrayList<>();

//前序遍历
public static List<Integer> preorderTraversal(TreeNode root) {

    if(root == null)
    {
        return null;
    }
    list.add(root.val);
    preorderTraversal(root.left);
    preorderTraversal(root.right);
    return list;
}
```

中序遍历

```
static List<Integer> list = new ArrayList<>();

//中序遍历
public static List<Integer> midorderTraversal(TreeNode root) {

    if(root == null)
    {
        return null;
    }
    midorderTraversal(root.left);
    list.add(root.val);
    midorderTraversal(root.right);
    return list;
}
```

后序遍历

```
public static List<Integer> BheaderTraversal(TreeNode root) {

    if(root == null)
    {
        return null;
    }

    BheaderTraversal(root.left);
    BheaderTraversal(root.right);
    list.add(root.val);
    return list;
}
```

其实这三个遍历差不多,只是list的添加元素的代码位置不一样
所谓的前序,中序,后序就是中间节点的位置