**南京师范大学中北学院**

**《数据结构》**

**实**

**验**

**报**

**告**

**姓 名: 钟玮**

**学 号: 85213241**

**班 级: 852132**

**日 期: 4.22**

**二叉树的基本操作-构造与遍历**

1. **实验目的**

**1、掌握二叉树的定义与性质；**

**2、掌握二叉树的基本操作——构造和遍历，如先序法构造二叉树、递归先序遍历、递归中序遍历和递归后序遍历等。**

1. **实验内容**

**1、使用先序法构造一颗二叉树；**

**2、对二叉树进行递归先序遍历；**

**3、对二叉树进行递归中序遍历；**

**4、对二叉树进行递归后序遍历。**

1. **实验主要代码**

**可以截图并做简要解释，如：**

1. **使用先序法构造一颗二叉树**

**int Binary\_tree\_Create(binary\_tree &T)**

**{**

**printf("开始创建\n");**

**int x;**

**scanf("%d", &x); // 从输入中读取一个整数**

**if (x == 0) // 如果读取的整数是0，那么创建一个空节点**

**T = NULL;**

**else**

**{**

**// 为新的节点分配内存**

**T = (binary\_tree\_node \*)malloc(sizeof(binary\_tree\_node));**

**if (!T)**

**return -2;**

**T->data = x; // 将读取的整数赋值给新节点的数据域**

**Binary\_tree\_Create(T->lchild); // 递归地创建左子树**

**Binary\_tree\_Create(T->rchild); // 递归地创建右子树**

**}**

**return true;**

**}**

1. **对二叉树进行递归先序遍历**

**void PreOrdertraverse(binary\_tree T)**

**{**

**if (T)**

**{**

**printf("%d", T->data);**

**PreOrdertraverse(T->lchild);**

**PreOrdertraverse(T->rchild);**

**}**

**}**

1. **对二叉树进行递归中序遍历**

**void InOrdertraverse(binary\_tree T)**

**{**

**if (T)**

**{**

**InOrdertraverse(T->lchild);**

**printf("%d", T->data);**

**InOrdertraverse(T->rchild);**

**}**

**}**

1. **对二叉树进行递归后序遍历**

**void PostOrdertraverse(binary\_tree T)**

**{**

**if (T)**

**{**

**PostOrdertraverse(T->lchild);**

**PostOrdertraverse(T->rchild);**

**printf("%d", T->data);**

**}**

**}**

1. **main函数**

**int main()**

**{**

**binary\_tree T;**

**Binary\_tree\_Create(T);**

**printf("递归先序遍历：");**

**PreOrdertraverse(T);**

**printf("\n");**

**printf("递归中序遍历：");**

**InOrdertraverse(T);**

**printf("\n");**

**printf("递归后序遍历：");**

**PostOrdertraverse(T);**

**printf("\n");**

**return 0;**

**}**

1. **实验结果**

**实验运行结果的截图，如main函数运行结果截图：**

****