y

**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: 4.29**

**二叉树遍历的应用**

1. **实验目的**

**1、掌握二叉树的定义与性质；**

**2、掌握二叉树的非递归中序遍历及遍历的相关应用，如求二叉树结点个数、求二叉树叶子结点个数和求二叉树的深度等。**

1. **实验内容**

**1、对二叉树进行非递归中序遍历；**

**2、求二叉树结点个数；**

**3、求二叉树叶子结点个数；**

**4、求二叉树的深度。**

1. **实验主要代码**

**可以截图并做简要解释，如：**

1. **对二叉树进行非递归中序遍历**

**void NInOrderTraverse(BiTree T){**

**BiTree S[100],p;**

**int top=0;**

**if(T){**

**S[top]=T;**

**top++;**

**}**

**while(top!=0){**

**while(S[top-1]){**

**S[top]=S[top-1]->lchild;**

**top++;**

**}**

**top--;**

**if(top!=0){**

**p=S[top-1];**

**top--;**

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

**S[top]=p->rchild;**

**top++;**

**}**

**}**

**}**

**2、求二叉树结点个数**

**int NodeCount(BiTree T){**

**if(T==NULL)**

**return 0;**

**else**

**return NodeCount(T->lchild)+NodeCount(T->rchild)+1;**

**}**

1. **求二叉树叶子结点个数**

**int LeafCount(BiTree T){**

**if(!T)**

**return 0;**

**else if(!T->lchild && !T->rchild)**

**return 1;**

**else**

**return(LeafCount(T->lchild)+LeafCount(T->rchild));**

**}**

1. **求二叉树的深度**

**int GetHeigt(BiTree T){**

**if(!T)**

**return 0;**

**else{**

**int m,n;**

**m=GetHeigt(T->lchild);**

**n=GetHeigt(T->rchild);**

**return m>n?++m:++n;**

**}**

**}**

**5、main函数**

**void main(){**

**BiTree T;**

**int nodes,leafs,heigt;**

**printf("递归先序创建二叉树（999代表空树）:\n");**

**PreCreateBiTree(T);**

**printf("非递归中序遍历：\n");**

**NInOrderTraverse(T);**

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

**nodes=NodeCount(T);**

**printf("该二叉树的结点个数为:%d\n",nodes);**

**leafs=LeafCount(T);**

**printf("该二叉树的叶子结点个数为:%d\n",leafs);**

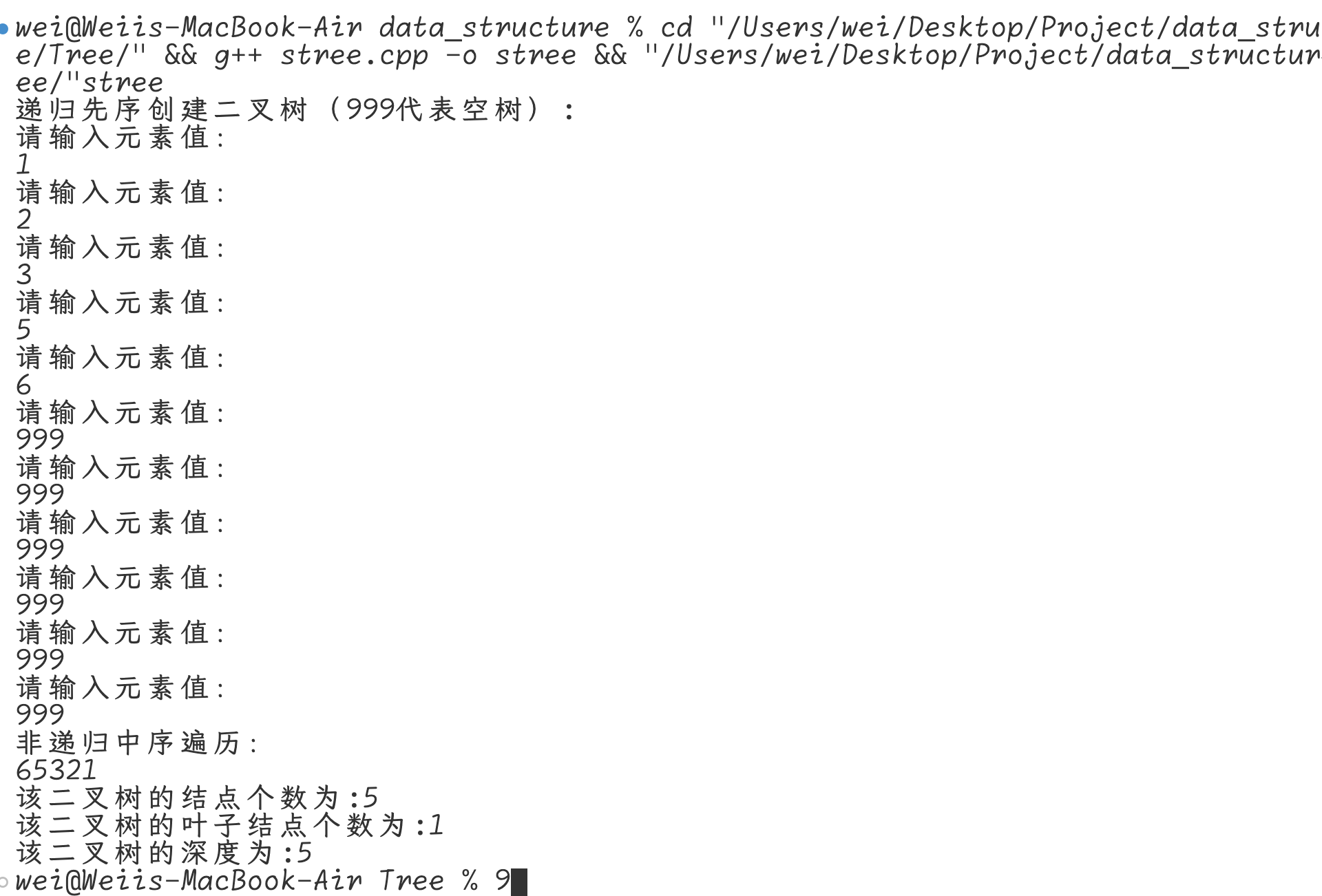
**heigt=GetHeigt(T);**

**printf("该二叉树的深度为:%d\n",heigt);**

**}**

1. **实验结果**

**实验运行结果的截图，如main函数运行结果截图：**

****