**《数据结构综合设计》实验报告**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 学院 | 人工智能与大数据学院 | 专业 | 虚拟现实技术 | 班级 | 21级3班 | 学生姓名 | 王怡佳 |
| 实验  周次 | 5-6 | 实验  日期 | 2023.4.6 | 学时 | 4 | 教师姓名 | 李昊康 |
| 项目名称 | | 树与二叉树的实现及应用 | | | | | |
| 实验  类别 | 🗹验证型实验 🞎设计型实验 🞎综合型实验 🞎其它 | | | | | 成绩：93 | |
| 1. 实验目的及具体要求   实验目的：   1. 掌握二叉树的特点及其存储方式。 2. 掌握二叉树的创建和显示方法。 3. 复习二叉树遍历的概念，掌握二叉树遍历的基本方法。 4. 掌握求二叉树的叶子节点数、树的总结点数和树的深度等基本算法。   具体要求：  用先序法建立一颗二叉树，并能按照广义表表示法显示二叉树结构。  编写先序遍历、中序遍历、后序遍历程序。  编写求二叉树结点数、树的总结点树和深度的程序。  设计选择式菜单，以选择菜单的方式进行操作。   1. 实验仪器、设备和材料   装有并能运行VS2019的电脑。   1. 实验内容、步骤及实验数据记录   **先序法建立二叉树代码**  #include <stdio.h>  #include <stdlib.h>  #include<iostream>  #define OK 1  using namespace std;  #define ElemType char  typedef struct BiTNode {  char data;  struct BiTNode\* lchild, \* rchild;  }BiTNode, \* BiTree;  **先序建立二叉树**  BiTree CreateBiTree() {  char ch;  scanf\_s("%c", &ch);  BiTree T= nullptr;  if (ch == '#')T = NULL;  else {  T = (BiTree)malloc(sizeof(BiTNode));  T->data = ch;  T->lchild = CreateBiTree();  T->rchild = CreateBiTree();  }  return T;//返回根节点  }  **//先序遍历二叉树**  void PreOrderTraverse(BiTree T) {  if (T) {  cout << T->data << " ";  PreOrderTraverse(T->lchild);  PreOrderTraverse(T->rchild);  }  }  **//中序遍历**  void InOrderTraverse(BiTree T) {  if (T) {  InOrderTraverse(T->lchild);  cout << T->data << " ";  InOrderTraverse(T->rchild);  }  }  /**/后序遍历**  void PostOrderTraverse(BiTree T) {  if (T) {  PostOrderTraverse(T->lchild);  PostOrderTraverse(T->rchild);  cout << T->data << " ";  }  }  int main() {  BiTree T;  T = CreateBiTree();//建立  cout << "先序遍历结果：" << endl;  PreOrderTraverse(T);  cout << endl;  cout << "中序遍历：" << endl;  InOrderTraverse(T);//输出  cout << endl;  cout << "后序遍历：" << endl;  PostOrderTraverse(T);  cout << endl;  system("pause");  }  **树的深度;**  int Depth(BiTree T)  {  int m,n;  if(!T)  return 0;  if(!T->lchild && !T->rchild)  return 1;  else  {  m = Depth(T->lchild);  n = Depth(T->rchild);  return 1+(m>n?m:n);  }  }  **树的结点**；  int Count(BiTree T)  {  int m,n;  if(!T)                                                      return 0;  if(!T->lchild && !T->rchild)  return 1;  else  {  m = Count(T->lchild);  n = Count(T->rchild);  return m+n+1;  }  }   1. 实验结果及分析   **先序法建立二叉树**      **运行结果：**    **树的深度：**    **树的结点：** | | | | | | | |

说明：1. 实验周次：填写实际上课周，如第5-8周上课填“5-8”或第10周上课填“10”。

1. 实验报告各部分内容需详实填写，按实验指导书上的评分标准给出分数。
2. 实验类型参考实验类型说明文件。