

《数据结构》实验



实验二 树型结构及其应用

主讲教师：王玮

实验教师：杨扬

计算机学院
哈尔滨工业大学（深圳）

实验内容

题目1

按层次遍历序列建立二叉树，并输出该树的前序、中序和后序遍历序列。

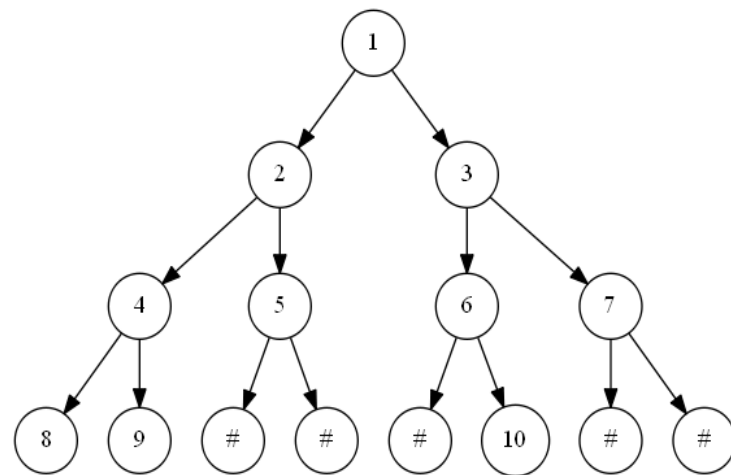
输入

- 15
- 1 2 3 4 5 6 7 8 9 # # # 10 # #

注：#表示空，输入可能不为满二叉树

输出

- preOrderTraverse is: 1 2 4 8 9 5 3 6 10 7
- inOrderTraverse is: 8 4 9 2 5 1 6 10 3 7
- postOrderTraverse is: 8 9 4 5 2 10 6 7 3 1



提示：可以使用队列来实现层序遍历构建二叉树。

注：只输入一个#时，需要返回空二叉树（NULL）！

实验内容

题目2

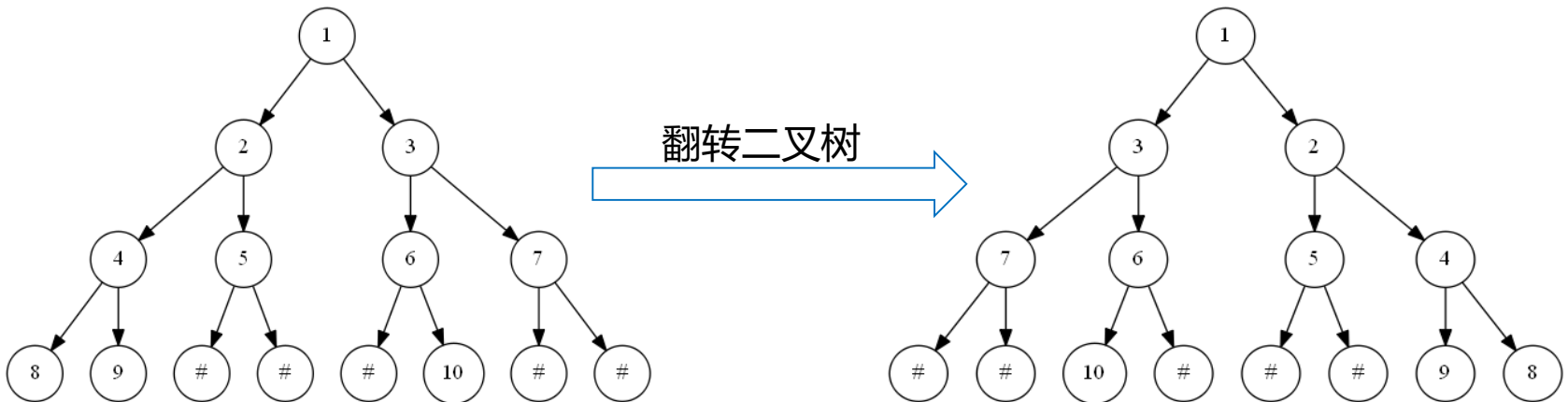
给定一棵二叉树，求取该树的镜像，即翻转该二叉树，输出中序遍历。

输入

- 15
- 1 2 3 4 5 6 7 8 9 # # # 10 # #

输出

- inOrderTraverse of invert tress is: 7 3 10 6 1 5 2 9 4 8



实验内容

题目3

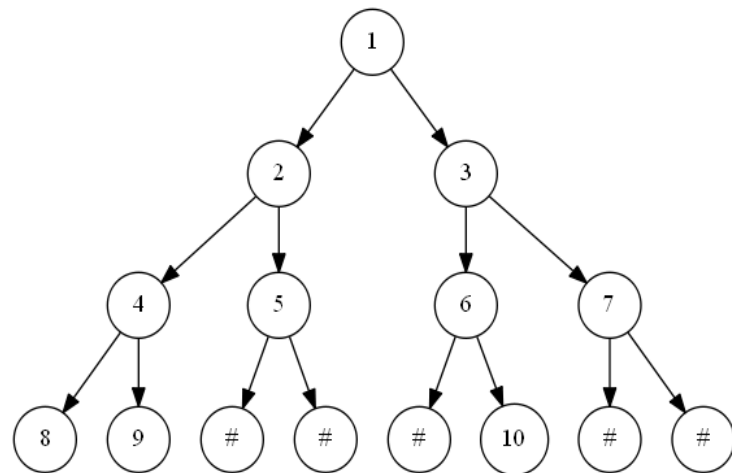
给定一棵二叉树，求取该二叉树的所有左子叶权重之和。

■ 输入

- 15
- 1 2 3 4 5 6 7 8 9 # # # 10 # #

■ 输出

- 8



(左孩子分别有2 4 6 8，但2 4 6都不是叶子结点，只有8才是左子叶，所以该树的左子叶权重之和为8)

注：只输入一个空二叉树时，输出为0。

实验内容

题目4

判断两个结点是否可达，如果不可达输出Too Far!!!，可达则输出起点到终点途经的所有结点(包含起点和终点)。

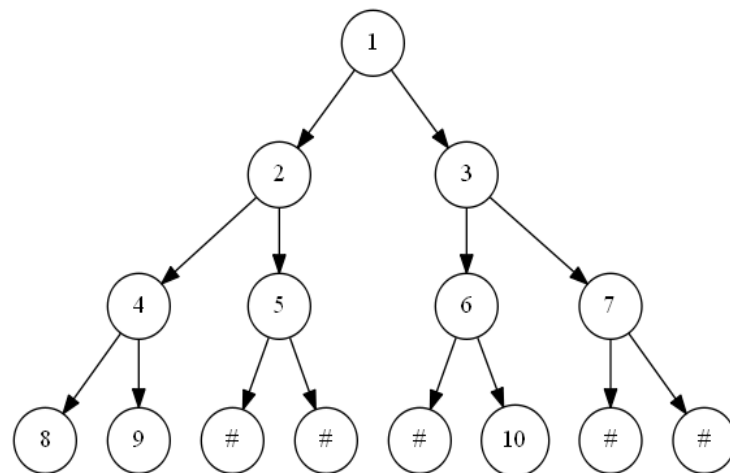
■ 输入

- 15
- 1 2 3 4 5 6 7 8 9 # # # 10 # #
- 2 9

■ 输出

- 2->4->9

注：本实验规定树是有向的，即结点不存在parent域，因此结点7到10是不可达的。



实验说明

- 参考代码**实现了队列封装**，可直接使用相关接口进行编码；
- 二叉树节点的数值，由输入的**char型**数值转换为**int型**数值，已经在void `getDigits(char *buff, int *data)` 中实现，可直接使用；
- 学习使用**递归**的方式解决问题。
- 感兴趣的同学可以自行学习二叉树可视化函数接口实现的代码，包括 `void createDotFile()`， `void plot()`。该部分需要下载graphviz 软件，并将main函数中的**`use_graphviz`**值修改为**`true`**后使用，详见任务说明word文档。**提交到OJ上的代码要将**`use_graphviz`** 设置为**`false`**，否则会报错。（非必选项，不计分）**

评分标准



- **总分100分**
- 层次遍历建二叉树及遍历 (40分)
- 翻转二叉树 (20分)
- 左子叶权重之和 (20分)
- 判断两个结点是否可达，并输出途径所有结点 (20分)

实验说明



Runtime Error

- 数组越界
- 内存泄露
- 变量没有初始化
- 函数在声明之前被调用