



# 南昌大学实验报告

学生姓名： 丁俊 学 号： 8003119100 专业班级： 信安 193 班

实验类型： ☒ 验证 ☐ 综合 ☐ 设计 ☐ 创新 实验日期： 5.6 实验成绩：         

## 一、实验项目名称

二叉树

## 二、实验目的

学习并掌握二叉树的遍历和操作方法

## 三、实验任务

- 1、编写算法函数 void levelbintree(bintree t)，实现二叉树的层次遍历。
- 2、编写函数 bintree prelast(bintree t)和 bintree postfirst(bintree t)，分别返回二叉树 t 在前序遍历下的最后一个结点地址和后序遍历下的第一个结点地址。
- 3、试编写一个函数，将一棵给定二叉树中所有结点的左右孩子互换。

## 四、主要仪器设备及耗材

**Dec++5.15 windows10**

## 五、实验步骤

### 1、二叉树层次遍历

```
1. #include "bintree.h"
2. char *a = "ABC##D##E##F##";
3.
4. void levelbintree(bintree t) {
5.     bintree queue[100];
6.     int l = 0, r = 1;
7.     queue[0] = t;
8.     bintree tmp;
9.     while (l < r) {
10.         tmp = queue[l]; // 记录队首元素
11.         l++; // 出队并打印
12.         printf("%c ", tmp->data);
13.         if (tmp->lchild) {
14.             queue[r] = tmp->lchild;
15.             r++;
16.         }
17.         if (tmp->rchild) {
18.             queue[r] = tmp->rchild;
19.             r++; // 入队
20.         }
21.     }
22. }
23.
24. int main() {
25.     bintree t;
26.     t = creatbintree();
27.     printf("二叉树的层次序列为:\n");
28.     levelbintree(t);
29.     preorder(t);
30.     puts("");
31.     postorder(t);
32.     return 0;
33.
34. }
```

**过程：**用数组模拟队列并且使用队列记录树的各个结点，类似于宽度优先搜索 bfs 算法，一层一层遍历，每次从队列中取出一个元素并且记录其子元素并入队。

## 2、二叉树 t 在前序遍历下的最后一个结点地址和后序遍历下的第一个结点地址

```
1. #include "bintree.h"
2. char *a = "ABC##D##EF#G###";
3.
4. bintree prelast(bintree t) {
5.     bintree p = t;
6.     if (p) {
7.         while (p && p->lchild || p->rchild) {
8.             if (p->rchild)
9.                 p = p->rchild;
10.            else
11.                p = p->lchild;
12.        }
13.    }
14.    return p;
15. }
16.
17. bintree prefirst(bintree t) {
18.     bintree p = t;
19.     if (p) {
20.         while (p && p->lchild || p->rchild) {
21.             if (p->lchild)
22.                 p = p->lchild;
23.             else
24.                 p = p->rchild;
25.         }
26.     }
27.     return p;
28. }
29.
30. int main() {
31.     bintree t, p, q;
32.     t = creatbintree();
33.     q = prelast(t);
34.     p = prefirst(t);
35.     if (t != NULL) {
36.         printf("前序遍历最后一个结点为:%c\n", q->data);
37.         printf("后序遍历第一个结点为:%c\n", p->data);
38.     } else {
39.         printf("二叉树为空! ");
40.     }
41.     return 0;
42. }
```

**过程：**因为是找前序遍历的最后一个结点，而前序遍历的顺序是父节点->左子节点->右子节点，所以我们直接开始从右子节点找，若右子树为空，则反过来找左子节点，最后找到的结点无左右子树跳出循环返回 p 结点的地址。

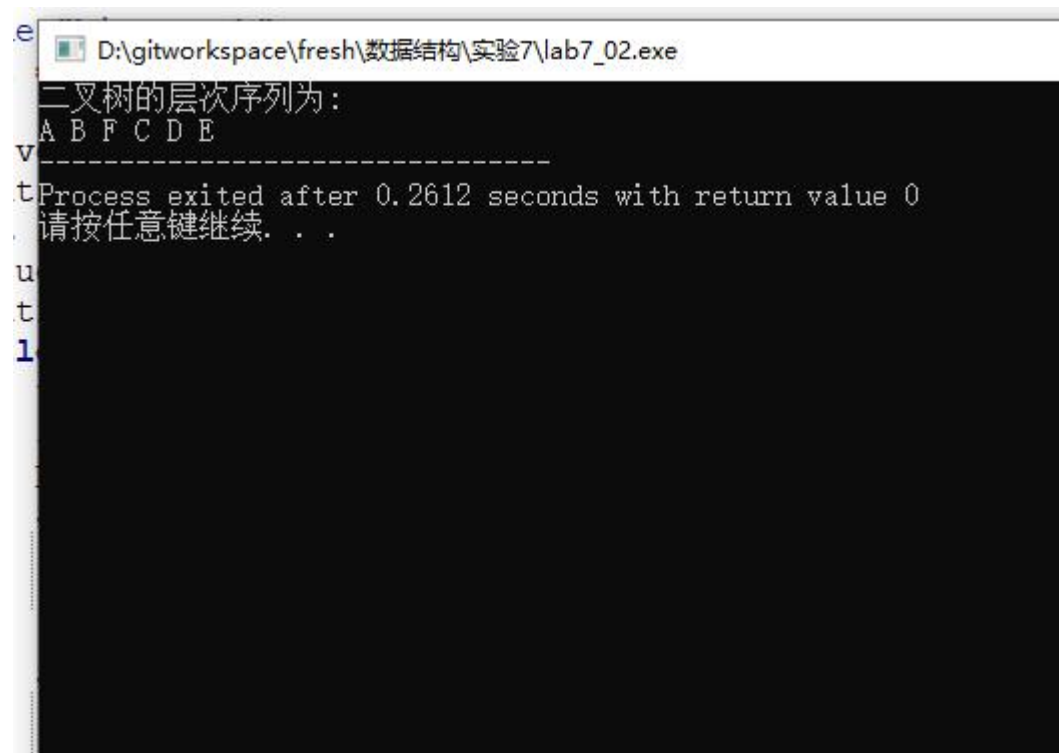
### 3、将一棵给定二叉树中所有结点的左右孩子互换。

```
1. #include "bintree.h"
2. char *a = "ABC##D##EF#G###";
3.
4. void change(bintree t) {
5.     bintree p;
6.     if (t == NULL)
7.         return;
8.     else {
9.         p = t->lchild;
10.        t->lchild = t->rchild;
11.        t->rchild = p;
12.        // 递归左子树
13.        change(t->lchild);
14.        // 递归右子树
15.        // 直到 t 为空返回 NULL
16.        change(t->rchild);
17.    }
18.
19. }
20.
21. int main() {
22.     bintree root;
23.     root = creatbintree();
24.     change(root);
25.     preorder(root);
26.     return 0;
27. }
```

**过程：**直接采取递归的方法，把每个结点(包括其本身)下面的儿子当作一个分支树，交换左右结点，再分别递归左右子树即可。

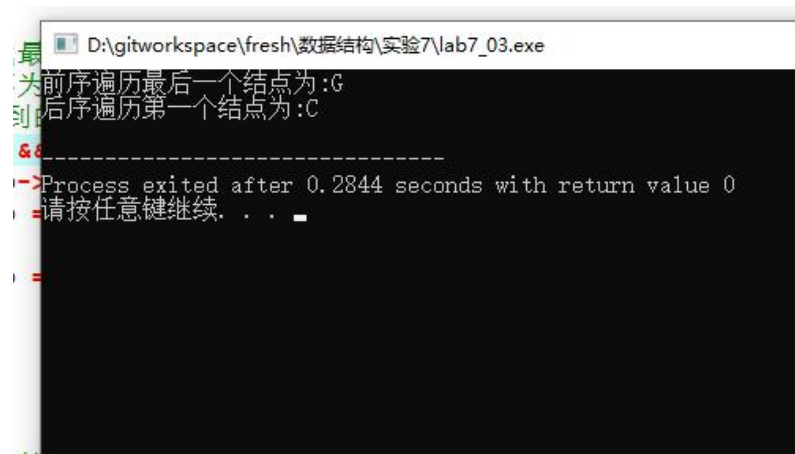
## 六、实验数据以及处理结果

1、



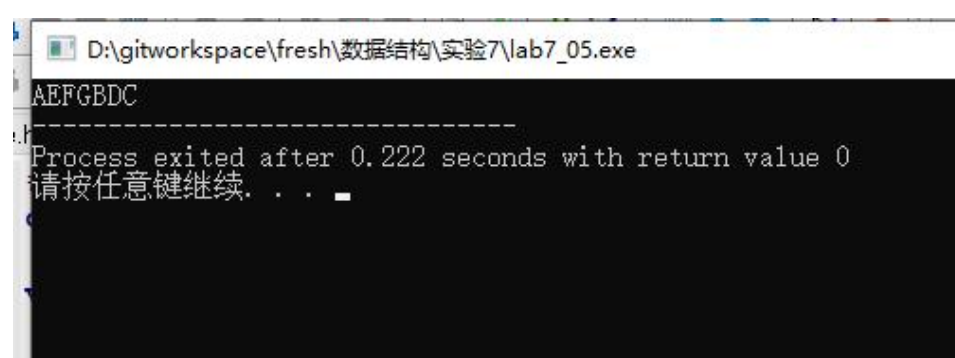
```
D:\gitworkspace\fresh\数据结构\实验7\lab7_02.exe
二叉树的层次序列为:
A B F C D E
-----
Process exited after 0.2612 seconds with return value 0
请按任意键继续. . .
```

2、



```
D:\gitworkspace\fresh\数据结构\实验7\lab7_03.exe
前序遍历最后一个结点为:G
后序遍历第一个结点为:C
-----
Process exited after 0.2844 seconds with return value 0
请按任意键继续. . .
```

3、



```
D:\gitworkspace\fresh\数据结构\实验7\lab7_05.exe
AEFGBDC
-----
Process exited after 0.222 seconds with return value 0
请按任意键继续. . .
```

## 七、思考讨论题或体会或对改进实验的建议

通过这次实验我更加熟练地掌握了树的结构和遍历的相关知识

## 八、参考资料

数据结构 C 语言版