**第五次作业要求**

1. **实现二叉链树，提供如下功能：**
2. createTree(char \*filename, T flag)，从文件读取结点来创新一棵二叉树；
3. 实现递归和非递归的二叉树前序、中序和后序遍历函数（非递归实现可等到下次课讲完再做）；
4. createTreeFromPreMid(T \* pre, T\* mid)，根据前序和中序遍历结果构建一个二叉树；
5. switch()，交换二叉树中所有结点的左右子树；
6. count2Degree()，统计二叉树中度数为2的结点个数；
7. checkFull()，判断二叉树是否是满二叉树；
8. checkComplete()，判断二叉树是否是完全二叉树；
9. **根据提供的测试数据分别构建不同深度的二叉树，进行如下测试**
10. 判断根据不同测试数据构建的二叉树是否是满二叉树或完全二叉树、统计二叉树中度为2的结点个数；
11. 测试不同深度的二叉树递归和非递归遍历算法在时间上的差异；