**二叉树辅助教学软件开发计划书**

|  |  |  |  |
| --- | --- | --- | --- |
| **项目名称** | 二叉树辅助教学软件 | **项目编号** | **Btree1** |
| **客户名称** | 在校学生及教师等学习或教授二叉树数据结构知识的人 | **客户代表** | 李智宇 |
| **编程人员** | 李智宇 | **工作量估算** | 7**±**1天 |
| **项目开始日期** | **2017-9-19** | **项目结束日期** | **2017-9-27** |

**一、需求分析**：

1. 软件功能：包含基本功能：创建二叉树、遍历（先序、中序、后序）二叉树、线索化（先序、中序、后序）二叉树、查找结点、查找叶子、插入结点、删除子树、求树高。
2. 软件要求：
   1. 软件基于真实的二叉树结构；
   2. 软件不依赖开发环境独立运行；
   3. 软件各个方面有严格的异常处理措施；
   4. 软件的流程、功能、界面、操作等逻辑清晰；
   5. 对软件流程、功能、界面、操作等的细节有充分考虑和处理。

**二、开发计划：**

二叉树方面：

1.我仅从图形方面观察到的情况：

先序遍历或者先序线索化，表现出多个三角形，其总的特点是，三角形群的尖端向心力指向是在二叉树总体的左下方，三角形群的开口处向心力指向总是在二叉树总体的右上方。

中序遍历或者先序线索化，表现出多个三角形，其总的特点是，三角形群的尖端向心力指向是在二叉树总体的正上方，三角形群的开口处向心力指向总是在二叉树总体的正下方。

先序遍历或者先序线索化，表现出多个三角形，其总的特点是，三角形群的尖端向心力指向是在二叉树总体的右下方，三角形群的开口处向心力指向总是在二叉树总体的左上方。

但是这三种规律性的画面，大致概括就是处于同一二维坐标系的基于原点（即向心点）发生变化而产生的坐标变化。不过这种规律可以对最后的结果进行验证，也可以作为加深自己对先序、中序、后序在图形中的直观印象。

1. 先序顺序为1根结点2前序遍历左子树3前序遍历右子树；中序顺序是1中序遍历左子树2根结点3中序遍历右子树；后序顺序是1后序遍历左子树2后序遍历右子树3根结点。相比第一种感官上的体会，第2种递归定义的逻辑结构更适合编程实现，所以暂时决定用第二种方案实现开发。

界面方面：

1. 设置menu菜单，类似之前加减乘除的时候，通过按键1234等由用户选择preoder、inoder等情况，如果是insert 等还需要等用户输入相应的参数。
2. 制作较为个性的开机logo，然后帮助文档，可参考默认示例和优秀示例。

二叉树图形显示方面：

这些方面还没有涉及，雏形可能比较依赖于示例以及提供的“C语言图形函数介绍.doc”文档。最后如果有时间的话会多做个性化尝试。

可能编写的相关函数（此处省略参数及返回值）：

CreateTree()：创建新的树。

DestroyTree()：销毁树。

ClearTree()：若树存在，将其清空。

IsEmpty()：判断该树是否为空树。

TreeDepth()：计算树的深度。

Root()：返回树的根结点。

Assign()：将树的某个已知结点赋为新值。

Parent()：返回某个结点的双亲。

TreeInsert()：在已知结点的左子树或右子树位置插入新结点。

TreeDelete()：删去该结点并删去其所有子树。

Preorder()：先序遍历。

Inorder()：中序遍历。

Postorder()：后序遍历。

Prelink()：先序线索化。

Indor()：中序线索化。

Postorder()：后序线索化。

Leaf()：找出所有叶子结点。