**《数据结构》实验报告X**

**学号：09022107 姓名：梁耀欣**

**实验题号：\_实验三\_BST\_\_\_ 实验日期：\_**2023.11.18**\_\_ 实验类型：** 必做

**1．问题描述：**描述实验内容和要求以及需要解决的问题。

**提示：**结合教师课堂讲授内容，仔细分析实验要求，对实验内容进行需求分析。

实验需求：实现二叉搜索树（BSTree）的类定义和成员函数定义，包括构造函数、析构函数、中序遍历、先序遍历、层次遍历、搜索、插入、清空、删除、以及输出树结构的函数并设计驱动函数 main()，通过伪随机函数生成一个由 n 个不重复字符（ASCII码在 [65, 90] 之间）组成的序列，并将其存入输入文件 in.txt，进行几个不同函数的测试。

**2．算法思想：**详细描述解决相应问题所需要的算法设计思想。

构建二叉搜索树：通过类的构造函数，从文件中读取一个由n个互不相同的key值组成的序列，然后逐一插入到二叉搜索树中，调用Insert函数实现。在每次插入后，通过PrintTree函数展示构建的过程。

插入节点：Insert函数通过递归实现，如果根节点为空，直接创建一个新节点作为根节点；否则，根据key值的大小关系递归地插入左子树或右子树。

遍历操作：

中序遍历：递归地访问左子树，访问当前节点，再递归地访问右子树。

先序遍历：先访问当前节点，再递归地访问左子树和右子树。

层序遍历：利用队列按层次访问节点。

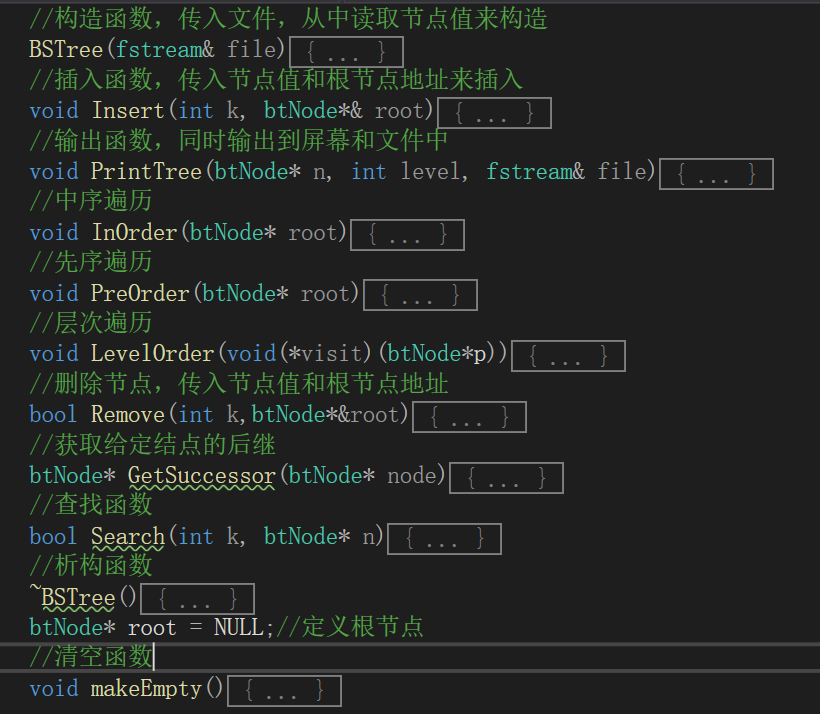
删除节点：通过递归实现，找到要删除的节点后，分三种情况处理：没有子节点：直接删除节点。有一个子节点：用子节点替代要删除的节点。有两个子节点：找到后继节点，用后继节点替代要删除的节点。

搜索节点：通过循环查找给定的key值，根据key值与当前节点的大小关系递归地在左子树或右子树中搜索。

后继节点获取：获取给定节点右子树中最左边的节点，处理连接关系，保持二叉搜索树的结构。

析构过程：根据先序遍历序列数组逐个按照先序删除节点，最终释放根节点的内存。

**3．功能函数：**描述所设计的功能函数。如果有多个函数，需要描述它们之间的关系。



**4．测试数据：**设计测试数据，或具体给出测试数据。

提示：要求测试数据能全面地测试所设计程序的功能。

**根据实验要求，在main函数中向数据文件“in.txt”中使用伪随机数生成一个由[65,90]之间n个字符组成的序列，进而调用各种功能函数来测试。**





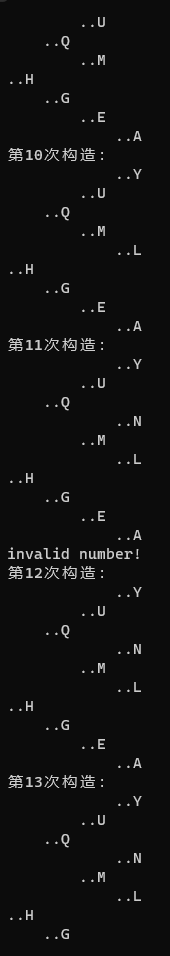
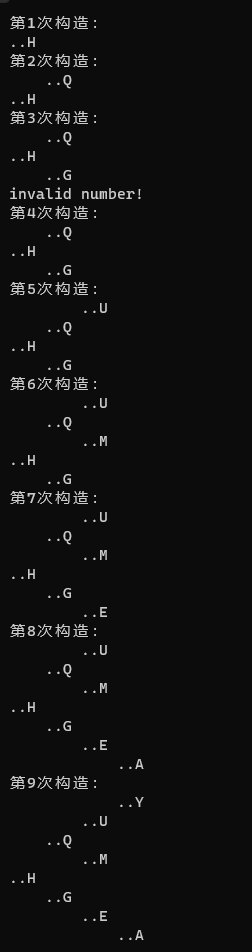


**由于程序自动调用析构函数，这里不做测试。**

**5．测试情况：**给出程序的测试情况，分析运行结果，显示实验结果截图。

**对程序进行测试情况如下：**

**·构造函数**



**伪随机数自动生成的n为20，因此进行了20次构造，由于二叉搜索树不能有重复值，在识别到重复值时进行了“invalid number”提示，最终构造完毕。**

**·搜索函数**

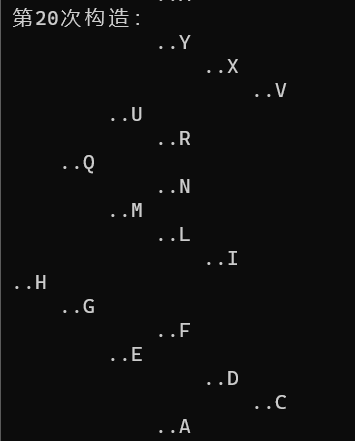


**搜索的输入值为68，经验证在输入序列中，输出found。**

**·三种遍历函数**

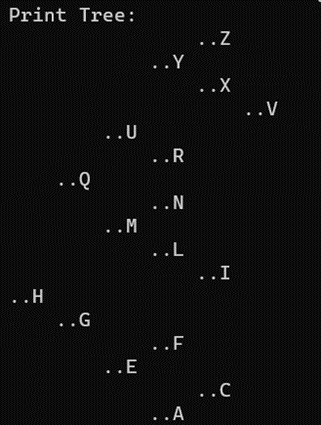
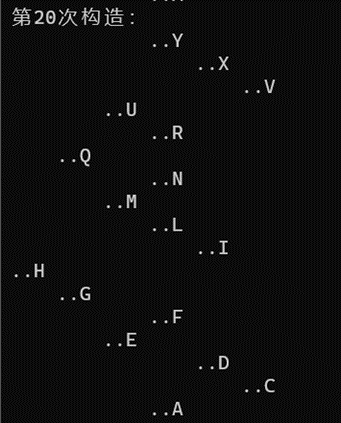


**·删除函数**



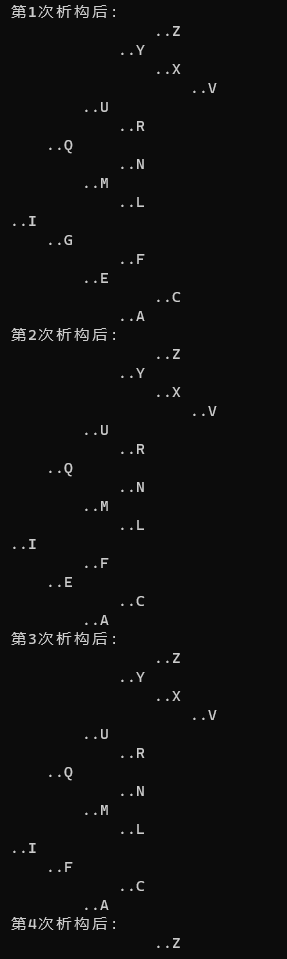
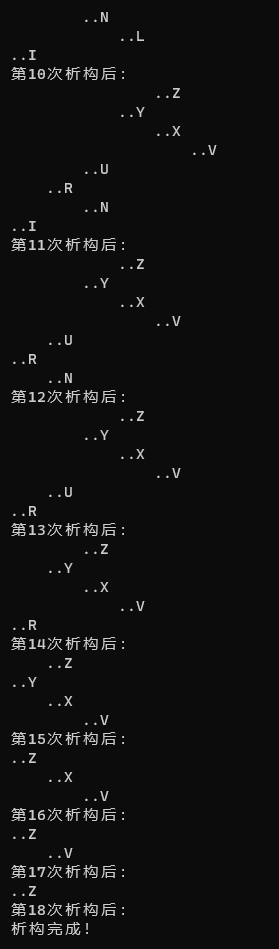
**对比删除前后的树形，发现成功删除‘D’。**

**·插入函数**

****

**对比插入前后的树形，发现成功插入‘Z’。**

**·析构函数**

**……**

**按先序序列输出析构过程如上。**

**6．实验总结：**写出实验过程中遇到的问题，以及问题的解决过程。分析算法的时间复杂度和空间复杂度，总结实验心得体会。

实验中遇到问题：没有仔细看测试函数的要求，一开始忽略了输出ASCII码，而且没有用先序析构，在最后检查的时候才发现，再进行了修改。另外的问题：对文件读写掌握不好，其中缺少了close()函数导致调试时间浪费，通过网上查资料发现。还有就是删除节点的时候遇到了双子节点的问题，在模拟删除子节点很多次后，找到了规律，重新定义了寻找后继的函数。

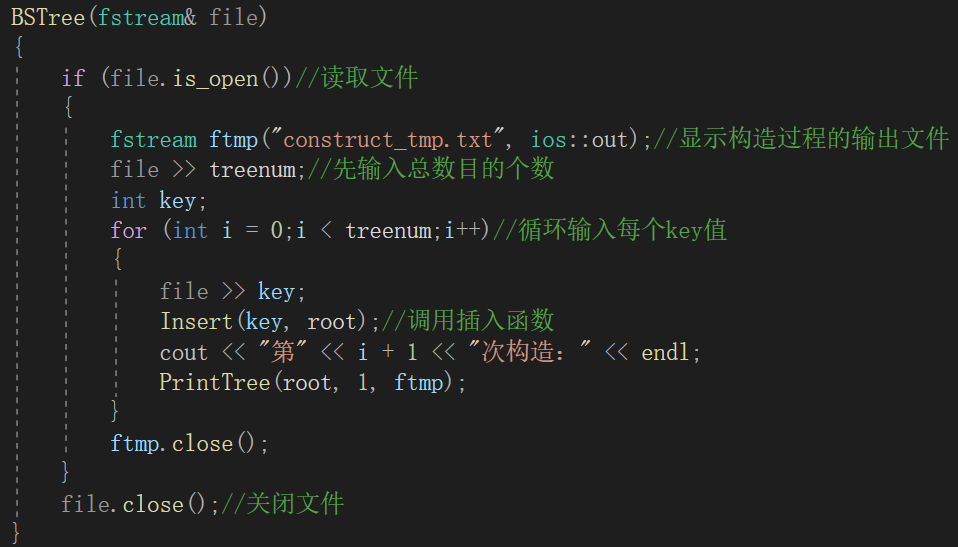
在平均情况下，插入、删除和查找的时间复杂度为 O(log n)，其中 n 是节点数目。在最坏情况下，树为斜树时，插入、删除和查找的时间复杂度为 O(n)。

空间复杂度主要取决于递归调用的栈空间，为 O(h)，其中 h 是树的高度。

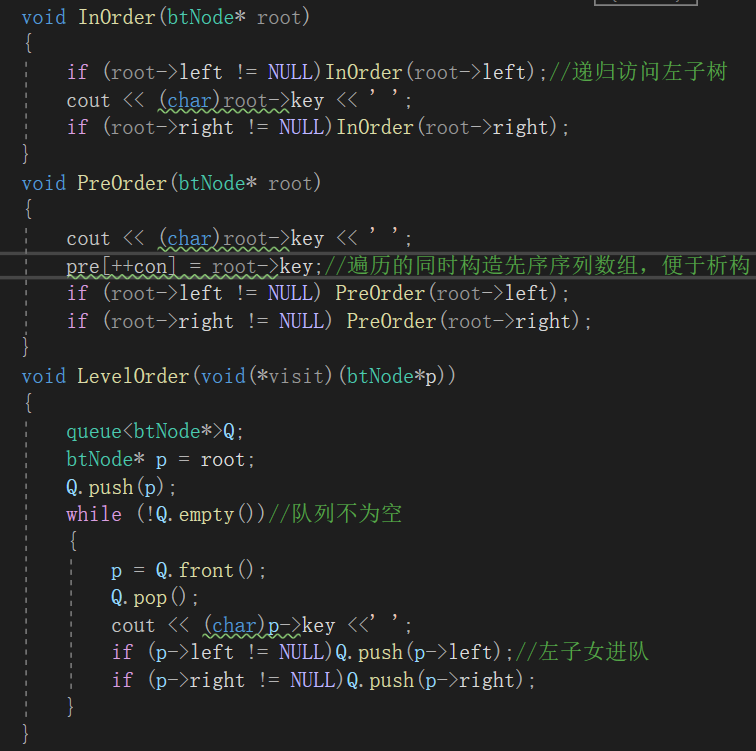
**7. 主要源代码：**

建议：源程序中应有充分的注释，例如注释每个函数参数的含义、函数返回值的含义、函数的功能、主要语句段的功能，等等。

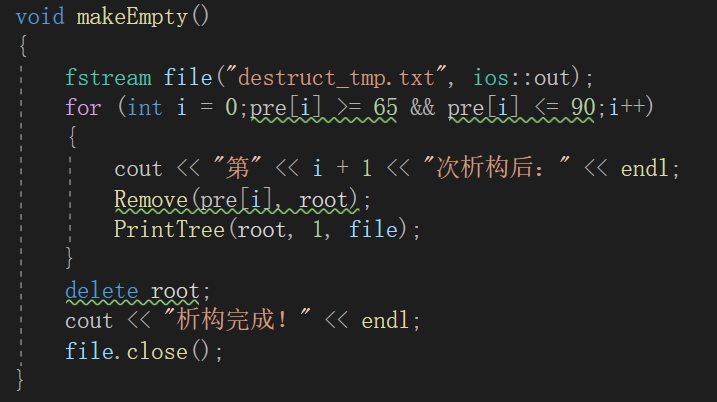
**构造函数：**



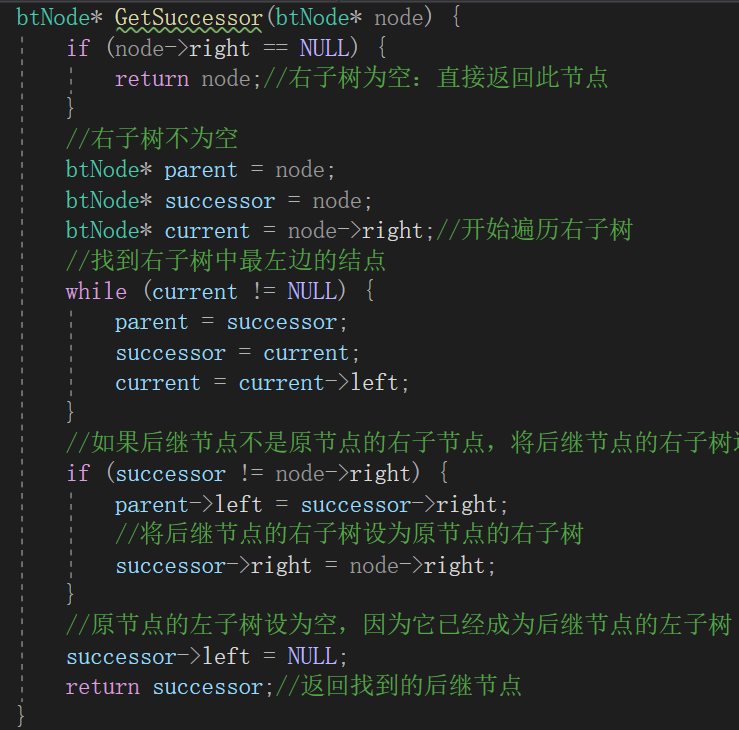
**三种遍历函数：**



**清空函数：**



**获取后继的函数：**



**8. 实验源程序：**附件清单

(1) Tree.cpp