**数据结构**

2022

**实 验 报 告**

|  |  |
| --- | --- |
| 实验项目名称： | 数据结构实验九 |
| 班级： | 21级9班 |
| 学号 | 2021302121185 |
| 姓名： | 陈聪睿 |
| 指导教师： | 董红斌 |
| 实验时间： | 2022.5.19 |

**实验一： 实现折半查找的算法**

**一、实验要求**



**二、实验环境**

**Visual studio 2019/ Windows平台**

**三、实验步骤及思路**

（1）题目分析与主要步骤

根据题目提示可知，需要创建一个折半查找算法。

使用非递归算法，用low、high存储查找的范围。

折半查找算法的基本步骤为：从当前顺序表的中部寻找元素，若没找到就从中间分为左右两部分，再从左右部分的中部查找元素；若找到即返回元素。

所以需要一个mid来计算中部元素在顺序表中的位置，公式为mid=（high+low）/2

（2）实验具体步骤截图

测试数据：题目已给出，结果如下图。

图片包含 文本

描述已自动生成

**四、实验结果及分析**

可以看出已经按照需要打印出了正确结果。

**五、总结**

可以看出，折半查找的效率较高，时间复杂度为O(log2n )

虽然折半查找的效率高，但要求查找表是按关键字有序的。

另外，折半查找需确定查找的区间，因此要求查找表的存储结构具有随机存取特性，所以只适用于顺序表，不适合于链式存储结构。

**实验二： 实现二叉排序树的基本运算方法**

**一、实验要求**



**二、实验环境**

**Visual studio 2019/ Windows平台**

**三、实验步骤及思路**

（1）题目分析与主要步骤

通过题目可知，需要创建插入二叉树结点函数InsertBST，创建二叉排序树CreateBST，删除二叉树某一结点的一系列函数DeleteChild、Delete、DeleteBST、查找二叉排序树某一元素（递归记录方法与非递归记录方法）SearchBST1、SearchBST2，输出二叉排序树DispBST、判断二叉排序树JudgeBST、销毁二叉排序树DestoryBST等方法。

二叉排序树与一般的二叉树类似，只有插入、删除、寻找时，因为二叉排序树的结构与一般二叉树不同，需要特殊注意。

I．插入结点时，可使用递归方法：

如果待插入位置不存在数据，直接插入。

如果待插入位置有数据，则需要比较。如果插入数据与原数据相同，则无需插入；如果比原数据小，则插入到左子树的某一位置；如果比原数据大，则插入到右子树的某一位置。

II．删除时，不能打乱原顺序，需要保持二叉排序树的结构。

首先调用DeleteBST函数，查找关键字k的位置，找到后调用Delete函数删除该节点，同时对其可能拥有的左右子树进行处理（若只有单个子树，就直接将其子树的根节点放置如原位置，若有两棵子树，调用DeleteChild函数，将其进行调整）。

III．查找目标结点，有递归与非递归两种记录方法。

递归记录方法较为容易，即查找目标元素，若没有的话去查找左右子树，存在的话直接输出该值，输出顺序为逆序。

非递归记录方法，查找方法同递归记录方法，但是为了让输出顺序为顺序，则需要使用一个路径数组进行记录。

（2）实验具体步骤截图

测试数据：题目已给出，结果如下图。

形状, 箭头

描述已自动生成

**四、实验结果及分析**

可以看出已经按照需要打印出了正确结果。

**五、总结**

利用树的特点存储数据，可以写出二叉排序树结构。

就平均时间性能而言，二叉排序树上的查找和折半查找差不多。但就维护表的有序性而言， 二叉排序树更有效，因为无须移动元素， 只需修改指针即可完成结点的插入和删除操作， 且其平均的执行时间均为O（log2 n)