重庆大学本科学生实验项目任务书

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 实验题目 | 二叉树与二叉查找树实践 | | | |
| 实验时间 |  | 实验地点 | |  |
| 实验性质 | □验证性 √设计性 □综合性 | | | |
| 实验目的  1. 掌握二叉树与二叉查找树的基本原理  2. 训练使用二叉树基本操作，通过编程解决不同难度问题的实践能力 | | | | |
| 实验内容   1. 二叉树与二叉查找树算法基本原理及时空复杂度分析方法 2. 在线编程解决3道不同难度的编程题，共40分   3． 在线编程时间（答题时长）150分钟，从开始答题时刻起算  4． 在线程序测评平台采用PTA (https://pintia.cn/)  5． 每人必须独立完成编程，可查阅教科书、PPT等资料，不得相互抄袭以及抄袭网上已有的程序  6． 实验课结束后，会对所有程序进行查重，如检测出有抄袭的程序，成绩计零分处理  注意：最后提交完整的实验报告，包括对每道题的算法思路、代码描述、复杂度分析等内容并回答思考题。 | | | | |
| 参考资料：   * Data Structures and Algorithm Analysis (C++ Version) Clifford A. Shaffer   + Introduction to Algorithms, 3rd Edition, MIT Press, T.H. Cormen, et al.   + 《数据结构（ C 语言版）》，严蔚敏，吴伟民编著，清华大学出版社 | | | | |
| 任务下达日期 2021 年 10月 29 日 | | | 完成日期 2021 年 11月 07 日 | |

说明：学院、专业、年级均填全称，如：计算机学院、计算机科学与技术、2021。

**《数据结构与算法》实验报告**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **年级、专业、班级** | | **2021计算机卓越一班** | | | **姓名** | **罗帆靖** |
| **实验题目** | 二叉树与二叉查找树实践 | | | | | |
| **实验时间** | **2021.11.07** | | **实验地点** | **线上** | | |
| **实验成绩** |  | | **实验性质** | **□验证性** √**设计性 □综合性** | | |
| 教师评价：  □算法/实验过程正确； □源程序/实验内容提交 □程序结构/实验步骤合理；  □实验结果正确； □语法、语义正确； □报告规范；  其他：  评价教师签名： | | | | | | |
| 实验目的  1. 掌握二叉树与二叉查找树的基本原理  2. 训练使用二叉树基本操作，通过编程解决不同难度问题的实践能力 | | | | | | |
| 二、实验项目内容  注：每道题按下面的格式分别描述  实验课题1：  题目内容：The Kth Largest in BST  解题思路：首先题目需要求BST中第K大的数字，我想到的是利用BST的中序遍历（变式，先RChild）来从大到小用遍历，并用count来计数。当遍历到count==k的节点时，向上Return此节点。  源代码（格式正确，对关键步骤加注释）：    时间与空间复杂度分析:  递归实现中序遍历，递归的本质是系统自动实现栈操作，系统栈需要记住每个节点的值，因此空间复杂度为O(N);  由于每一步的++count是常数复杂度，因此总体时间复杂度有以下公式：T(N)=2T(N/2)+O(1)，计算可知时间复杂度为O(N)  实验课题2：  题目内容：从下往上打印指定元素的所有祖先  解题思路： 利用递归的思想，深度优先算法向下搜索，类似先序遍历的方法，若找到节点data为ch的节点，则return返回调用其递归的栈（即其祖先）。在具体代码实现中，利用返回1或0来在不同栈中传递数据标志是否找到相应元素，若其左子树或右子树中有其一返回为1，则其一定是相应元素的祖先，打印即可，反之则否。  源代码（格式正确，对关键步骤加注释）：    时间与空间复杂度分析:  递归实现，系统自动建栈，每个元素都有其对应栈，因此空间复杂度为O(N);  对应的print操作为常数复杂度，因此时间复杂度为O(N)  实验课题3：  题目内容：构造二叉检索树  解题思路：先构造一个最外层的creatBST函数，在函数中循环将缓冲区的data存入临时变量a，再调用insertNode函数接口，不断循环直到输入了0再跳出函数，完成构建返回此BST。  在insertBST函数中，递归实现：若小于则向左子树走，若大于则向右子树走的过程，直到其指向空节点，意味着到达了需要构建节点的位置，在此实现节点的构造、初始化及接入BST。返回的true和false对应着构造的成功与否。  源代码（格式正确，对关键步骤加注释）：      时间与空间复杂度分析:  在最坏情况下，每次插入的值都比前者小或者大，则插入的时间复杂度为1+2+3+…n则为O(N),在平均情况下，每次插入所需的比较次数不会超过树的高度logN，则平均情况下时间复杂度为O(NlogN)  系统会自动对每次递归调用分配栈，而最多递归层数为logN则其空间复杂度为O(logN) | | | | | | |
| 三、思考题  遍历二叉树是按一定的顺序依次访问树中各结点并输出结点存放的数据。二叉树有四种遍历方式：前序、中序、后序遍历即层序遍历。假设二叉树各结点的数据互异。如果出现如下的情况，即对二叉树用两种不同的方式遍历，生成的结点序列却相同，则该二叉树具有什么特征？   1. 前序遍历与中序遍历相同   每个节点只有右孩子或者没有子节点   1. 后序遍历与中序遍历相同   每个节点只有左孩子或者没有子节点   1. 前序遍历与后序遍历相同   只有一个根节点   1. 前序遍历与层序遍历相同   1.只有左子树  2.最多只有根节点有左子树或者没有左子树 | | | | | | |