

华东师范大学数据科学与工程学院上机实践报告

课程名称：算法设计与分析

指导教师：金澈清

上机实践名称：二叉搜索树

上机实践编号：No.8

年级：22级

姓名：唐健峰

学号：10225501408

组号：408

上机实践成绩：

上机实践日期：

一、目的

- 1. 熟悉算法设计的基本思想
- 2. 掌握二叉树搜索的基本思想，并且能够分析算法性能

二、内容与设计思想

- 1. 编程实现二叉搜索树，能根据给定字符串（50,20,80,null,null,60,90,null,null,null,100）构建对应二叉搜索树，实现二叉搜索树的各种操作，包括先序遍历、取最小值、取最大值、搜索节点、获取前驱节点以及获取后驱节点。
- 2. 根据二叉搜索树的中序遍历，输出二叉搜索树的先序和后序遍历结果。
- 3. 选择合适的数据规模，计算二叉搜索树在不同数据量下查询操作的所需时间。
- 4. 思考题：对比二叉搜索树、顺序访问和散列表的性能。

三、使用环境

推荐使用C/C++集成编译环境。

四、实验过程

- 1. 写出算法的源代码；
- 2. 以合适的图来表示你的实验数据

第一二题可以看我的水杉.cpp源码bst里的方法；

```
#define RANGE 1000000 // 数字范围
#define INSERT 100000 // 插入值得个数
#define TIMES 10000000 // 寻找次数
```

RANGE	1000000							1000000						
INSERT	100000							10000						
TIMES	10	100	1000	10000	100000	1000000	10000000	10	100	1000	10000	100000	1000000	10000000
消耗时间	0.008ms	0.05ms	0.455ms	4.377ms	42.359ms	457.395ms	4315.48ms	0.004ms	0.038ms	0.471ms	4.591ms	40.786ms	316.853ms	3061.03ms

RANGE	1000000						
INSERT	1000000						
TIMES	10	100	1000	10000	100000	1000000	10000000
消耗时间	0.012ms	0.094ms	1.057ms	9.034ms	98.321ms	892.565ms	8557.3ms

五、总结

对上机实践结果进行分析，问题回答，上机的心得体会及改进意见。

- 1.二叉搜索树：对于二叉搜索树，从根节点开始遍历，按照左子树小于右子树的规则进行排序。因此，在查找单个元素时，二叉搜索树的查找效率较高，最好情况下的时间复杂度为 $O(\log N)$ ，最坏情况下的时间复杂度为 $O(N)$ 。
 - 2.顺序访问：顺序访问是按照顺序逐个比较查找目标，因此时间复杂度为 $O(N)$ 。在数据量较小的情况下，顺序访问是一种较为简单的查找方式，但是当数据量较大时，时间复杂度会显著增加。
 - 3.散列表：散列表是一种哈希表数据结构，它通过“键值”对来快速找到对应的“值”。散列表的查找效率较高，时间复杂度为 $O(1)$ ，但是散列表需要使用哈希函数对键值进行处理，存在哈希冲突等问题，需要进行额外处理。
- 综合来看，在单个元素查询的场景中，二叉搜索树的查找效率较高，而散列表的效率最高；在查找多个元素的场景中，若查询频率高，则散列表效率更高，否则二叉搜索树效率更优。而顺序访问相比于前两者，效率较低，不适合用于大量数据查询的场景。