

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

年级：22 级

上机实践成绩：

指导教师：金澈清

姓名：石季凡

上机实践名称：红黑树

学号：

上机实践日期：2023.5.3

10225501403

上机实践编号：No.9

组号：1-403

一、目的

1. 熟悉算法设计的基本思想
2. 掌握构建红黑树的方法

二、内容与设计思想

1. 编写随机整数生成算法，生成 S 到 T 范围内的 N 个随机整数并输出；
2. 编写红黑树构建算法，中序遍历各节点，输出颜色和值；
3. 随机生成 $1e2$ 、 $1e3$ 、 $1e4$ 、 $1e5$ 、 $1e6$ 个不同的数，使用红黑树构建算法，并画图描述不同数据量下的运行时间差异；
4. （思考题选做）对比红黑树和普通搜索二叉树在不同情况下插入和查找的性能差异。

三、使用环境

推荐使用 C/C++ 集成编译环境。

四、实验过程

1. 写出红黑树构建算法的源代码
2. 使用合适的图表表达你的实验结果

五、总结

各个函数的思路

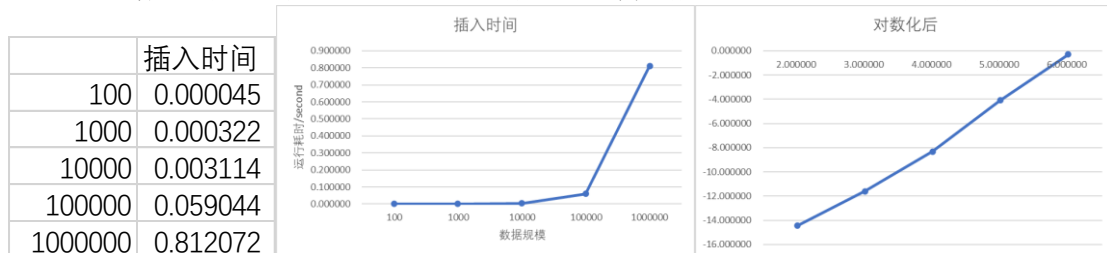
【插入】：大体过程与二叉搜索树类似，但是在插入后需要对结构进行调整，即优化其结构使其尽量为一个平衡二叉树，其中调整最多需要三次旋转，共有 8 种情况。

【旋转】：左旋，即将目标节点其 right 作为其 parent 的 right，目标节点的 right 为其 parent，目标节点的 parent 再替换为其 grand，而其 grand 的子代为目标节点，右选类似，即将目标节点其 left 作为其 parent 的 left，目标节点的 left 为其 parent，目标节点的 parent 再替换为其 grand，而其 grand 的子代为目标节点。

【搜索】：与二叉搜索树相同，即不断比较目标节点的 key 与 target 的 key，若大于则传递左子，反之右子。

【遍历】：即通过递归不断打印需要的变量。

【3】根据小问 3 的要求，我进行了实验，得到如下结果。插入时间的单位为秒。



由于初始数据不便于观察，我对数据规模取了以 10 为底的对数，对时间取了以 2 为底的对数。

由以上结果可见，由于红黑树优秀的结构，其不断插入后也可以维持一个相对平衡的二叉搜索树，即高度 h 较为稳定，因此当我们取对数后，我们可以发现，时间的变化很接近线性变化。

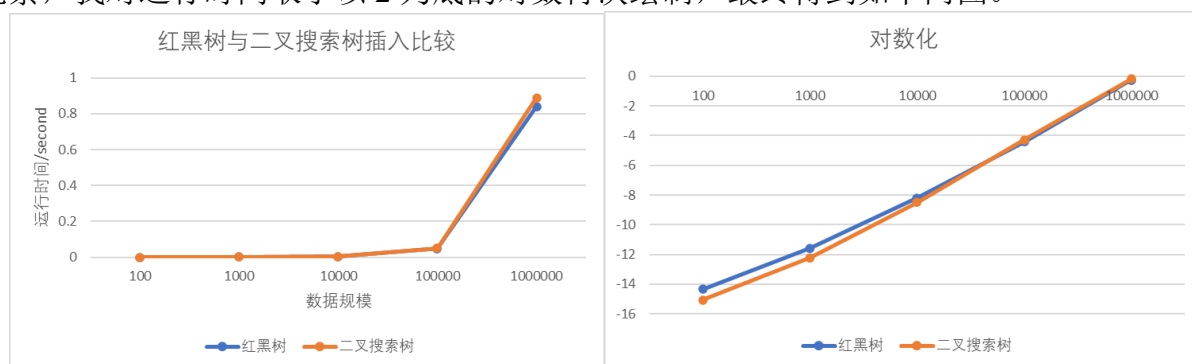
接下来我们来对红黑树与二叉搜索树进行比较，首先是插入，由于红黑树每一步的插入都需要进行旋转调整，而二叉搜索树则不需要，因此猜测来看，在数据量较小时，红黑树的插入时间会慢于二叉搜索树，但是当数据量大时，由于红黑树结构的优越性，时间会快于二叉搜索树。

【插入比较实验】下面进行插入实验，即分别插入数据量为 $1e2, 1e3, 1e4, 1e5, 1e6$ 的数据量，来进行比较。

实验得到的结果如下

	红黑树	二叉搜索树
100	0.000048	0.000029
1000	0.000326	0.00021
10000	0.003381	0.002774
100000	0.04704	0.05104
1000000	0.839666	0.889981

得到数据绘制为折线图如下，由于图像不易于观察，我对运行时间取了以 2 为底的对数再次绘制，最终得到如下两图。



实验的结果验证了我的猜想，即实验中，数据量小于 $1e5$ 时二叉搜索树快于红黑树，数据量大于 $1e6$ 时红黑树由于其平衡的结构，插入开始快于二叉搜索树，即证明了红黑树确实在结构上有优化。

【搜索实验】下面进行搜索实验，实验的已经插入的数据规模分别为 $1e3, 1e4, 1e5, 1e6, 1e7$ ，每次搜索 100 次，记录搜索时间，得到如下结果



其中搜索时间的单位为秒

由以上结果我们可以发现，无论是插入了多少数据，红黑树的搜索时间总是小于二叉搜索树，这也是二叉搜索树建树操作更复杂的目的，即让二叉树更接近一个平衡二叉树，同时由于红黑树最多只要旋转三次即可完成调整，这使其调整的耗时不会随着数据量增加而增加太多，因此在大规模的数据下，红黑树是在建树与搜索等操作用时都优于二叉搜索树的一个数据结构。

红黑树回答了上个实验中提出的问题，即优化树必须从其结构的建立入手。