

中国科学技术大学计算机学院
《算法基础》实验报告



实验题目：lab3_区间树和红黑树

学生姓名：胡毅翔

学生学号：PB18000290

完成日期：2020 年 12 月 25 日

计算机实验教学中心制

2019 年 09 月

实验目的

- 1.实现红黑树和区间树的数据结构。
- 2.实现红黑树和区间树的插入，删除，左旋，右旋等算法。
- 3.对获得的实验数据(运行时间)进行分析，并与理论进行比较。

实验原理

本次实验所实现的算法有:红黑树的插入，删除，遍历；区间树的插入，删除，遍历查找等。其正确性已在《算法导论》一书中得到证明。具体实现见对应源文件目录(/src)下的头文件(RBT.h 和 interval.h)。

实验环境

- 1.PC 一台
- 2.Windows 系统
- 3.gcc 编译器

实验过程

目录框架

本次实验的目录框架如下图所示：

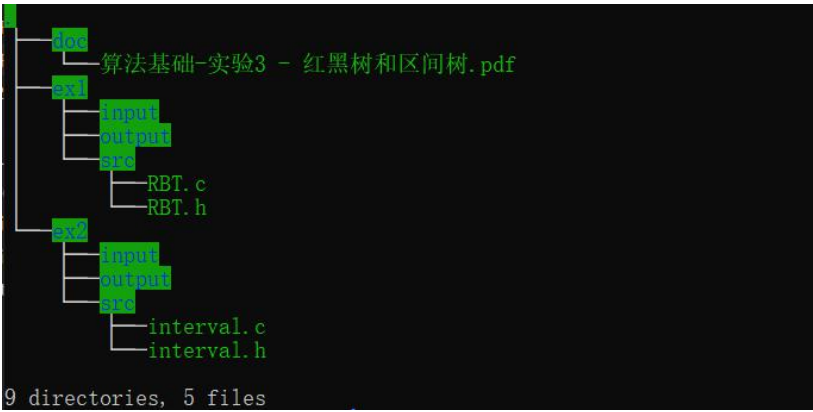


图 1

程序执行

执行 RBT.exe:

- 1.生成随机数。
- 2.建一个有 $n=20$ 个节点的树，并记录 $n/4=5$ 个节点指针(待删除)。

- 3.记录建树时间。
- 4.输出中序遍历结果。
- 5.删除 $n/4$ 个节点，输出删除的节点。
- 6.记录删除时间。
- 7.若仍有输入转 1，否则结束程序。
- 8.输出删除后的中序遍历序列。
- 9.令 $n=40, 60, 80, 100$ ，重复执行 2-8。

执行 interval.exe:

- 1.随机生成 30 个满足条件的区间。
- 2.建树。
- 3.插入节点，选取待删除节点。
- 4.中序遍历区间树。
- 5.删除节点。
- 6.生成 3 个符合条件的区间，在区间树上查找。

结果分析

运行时间统计分析

红黑树插入运行时间统计:

| 输入规模 | $T/\mu s$ |
|------|-----------|
| 20 | 128.1 |
| 40 | 10.4 |
| 60 | 11.8 |
| 80 | 13.7 |
| 100 | 17.6 |

表 1

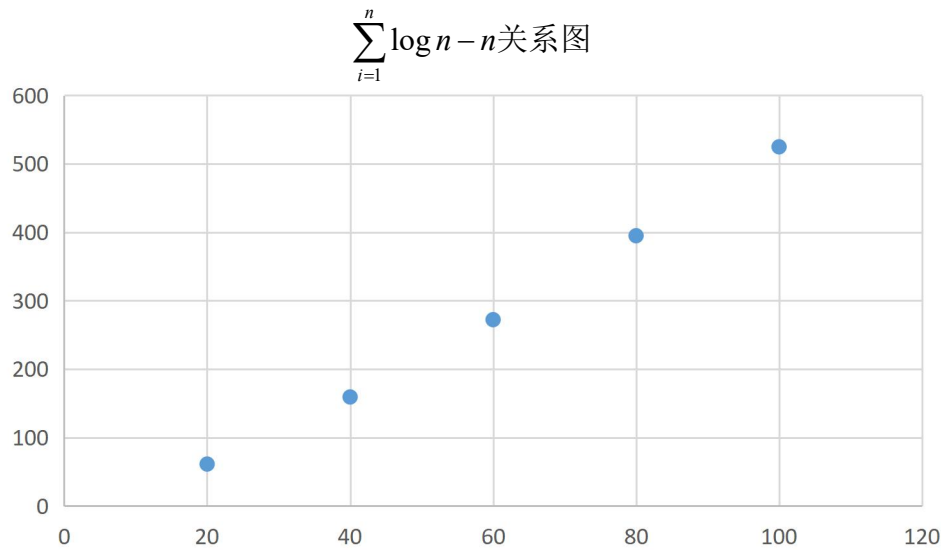


图 2

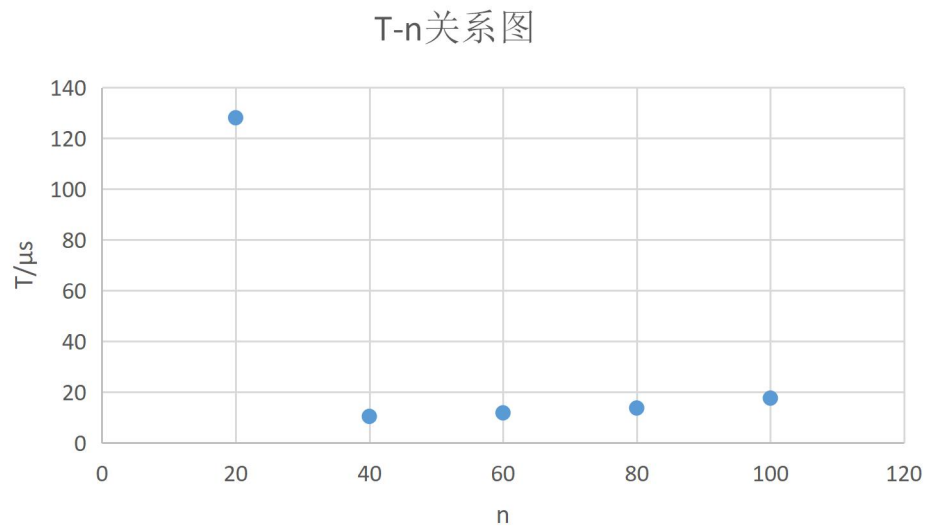


图 3

可以看出统计结果，与理论时间复杂度较大。误差分析如下：

N=20 时，运行时间远大于其他情况运行时间的可能原因是，在程序中第一次使用红黑树插入的相关代码，这些代码还在较低级存储中，需要花费较多时间加载这些代码。而 **N=40,60,80,100** 时，相关代码已在高级存储中，属于热代码，访存的时间开销减小了，运行时间较 **N=20** 时显著下降。

N=40-100 时，运行时间的增长关系与理论大致一致(大致为线性关系)，但可能其他操作占用的时间开销在 **n** 较小时，与插入操作的运行时间数量级相当，故运行时间的变化并不明显。比如，在本次实验中，插入操作过程中还包含了记录待删除节点等操作，这些操作可能对运行时间结果产生了较大影响。若要进一步分析，可以使时间记录更精确或者增大数据规模。

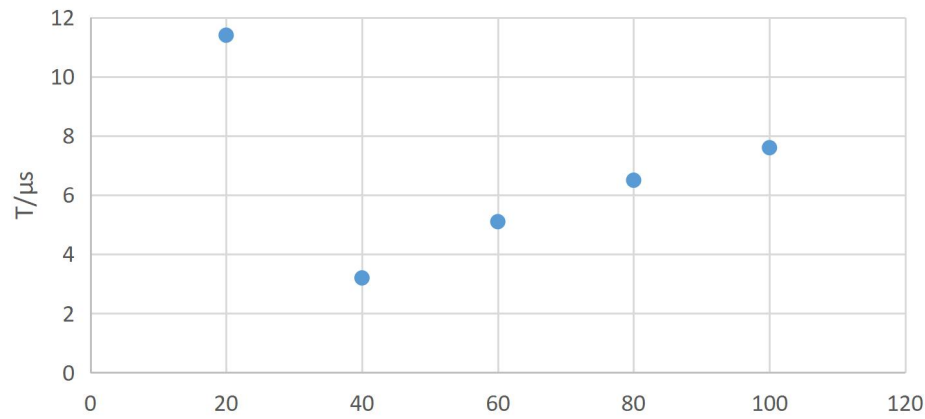
删除操作运行时间统计：

| n | $T / \mu s$ |
|-----|-------------|
| 20 | 11.4 |
| 40 | 3.2 |
| 60 | 5.1 |

| | |
|-----|-----|
| 80 | 6.5 |
| 100 | 7.6 |

表 2

T-n关系图



$\log n - n$ 关系图

图 4

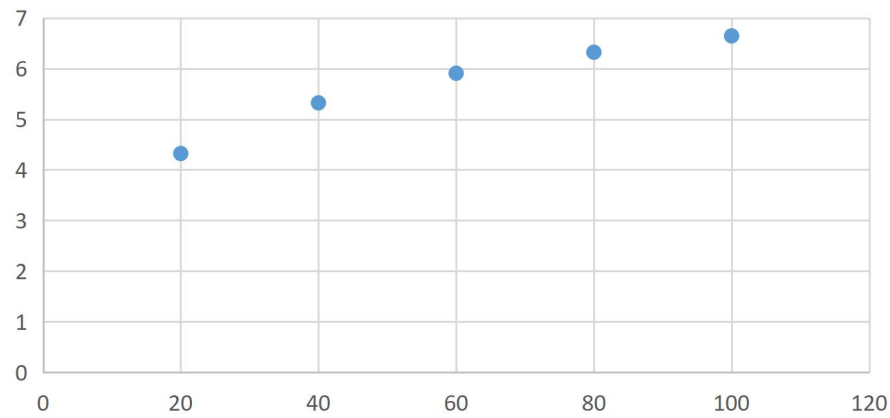


图 5

与插入操作一样，在 $N=20$ 时，运行时间最大，其他情况与理论时间复杂度相符。可能的原因仍是，在删除代码由“冷”变“热”过程中的访存时间代价较大。