

实验五 优先队列及排序

【实验目的】

- (1) 掌握优先队列及相关操作
- (2) 掌握各种排序算法

【实验内容】

1. 运用教材提供的 `BinaryHeap` 类，编写一个程序输入 N 个元素并
 - (1) 将它们一个一个地插入到一个堆中。
 - (2) 以线性时间建立一个堆。
2. 设二叉堆用显式链表示。考虑将二叉堆 lhs 和 rhs 合并的问题。假设这两个二叉堆均为满的完全树，分别包含 2^l-1 和 2^r-1 个节点。(选做)
 - (1) 若 $l=r$ ，给出合并这两个堆得 $O(\log N)$ 算法
 - (2) 若 $|l-r|=1$ ，给出合并这两个堆得 $O(\log N)$ 算法。
 - (3) 给出一个简单算法来找出位于位置 i 上的树节点。
3. 比较不同排序算法的执行情况：

运用选择排序、冒泡排序、插入排序、希尔排序、归并排序、快速排序、桶排序和基数排序算法，分别编写程序对 N ($N = 100, 1000, 5000, 10000, 50000, 100000$) 个随机整数进行排序，并且在表 1 中记录这些算法的运行时间。

提示 1：必须保证不同的算法有相同的输入数据序列。例如， $N=100$ ，所有的排序算法应该排序相同的 100 个整数。运行时间不包括产生 N 个随机整数。另外，请说明你的运行环境，包括 CPU、RAM 和操作系统。

提示 2：检测一个 JAVA 程序运行时间的 JAVA 代码：

```
long startTime = System.currentTimeMillis(); //获取当前时间
//doSomething();    //要运行的 java 程序
long endTime = System.currentTimeMillis();
System.out.println("程序运行时间: "+(endTime-startTime)+"ms");
```

表 1 排序执行情况

N	100	100000
Selection sort			
.....			

Bucket sort			
-------------	--	--	--