**实 验 报 告**

|  |  |
| --- | --- |
| **课程名称：** | **数据结构与算法分析** |
| **学生姓名：** | **叶子繁** |
| **学生学号：** | **201730684427** |
| **学生专业：** | **软件工程** |
| **开课学期：** | **2018-2019第一学期** |

**软件学院**

**2018年10月**

# **内排序算法的实现及分析**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **地 点：** | | | B7 楼 | | 231 房； | | **实验台号：** | | 87 |
| **实验日期与时间：** | | | 2018.11.9 | | | | **评 分：** | |  |
| **预习检查纪录：** | | |  | | | | **实验教师：** | |  |
| **电子文档存放位置：** | | | www.github.com/cinkye/Javastudy | | | | | | |
| **电子文档文件名：** | | | Exp3\sort.cpp | | | | | | |
| **批改意见：** | | |  | | | | | | |
| 出勤情况 | 实验报告质量 | | | | | |
| 任务完成情况 | | 源代码注释 | | 报告内容 | |
| □ 出勤  □ 缺勤  □ 早退  □ 请假 | □ 完成  □ 未完成 | | □ 优秀  □ 良好  □ 通过  □ 未通过 | | □ 优秀  □ 良好  □ 通过  □ 未通过 | |

# 报告内容

1、实验目标

2、实验环境

Ubuntu 18.04, Windows 10, Visual Studio Code, Code::Blocks

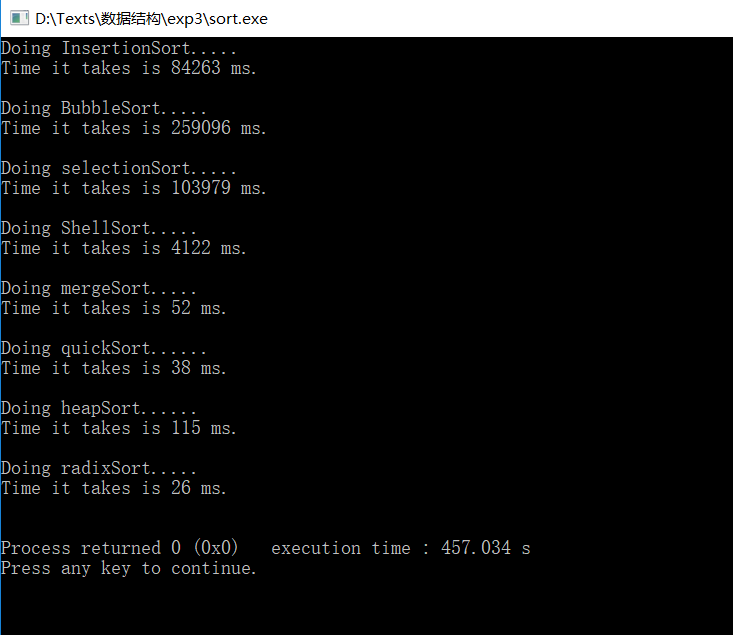
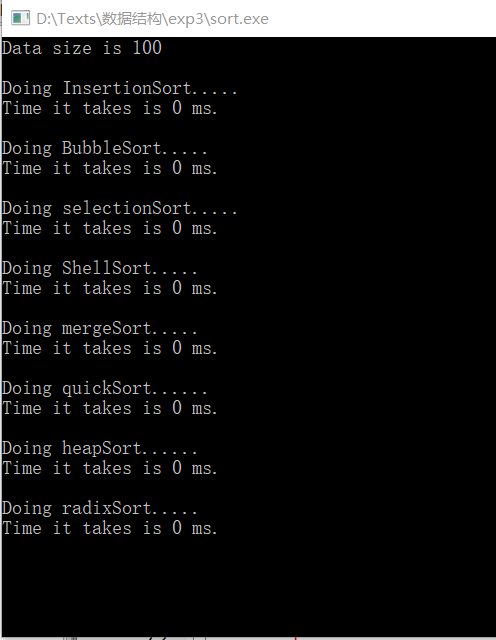
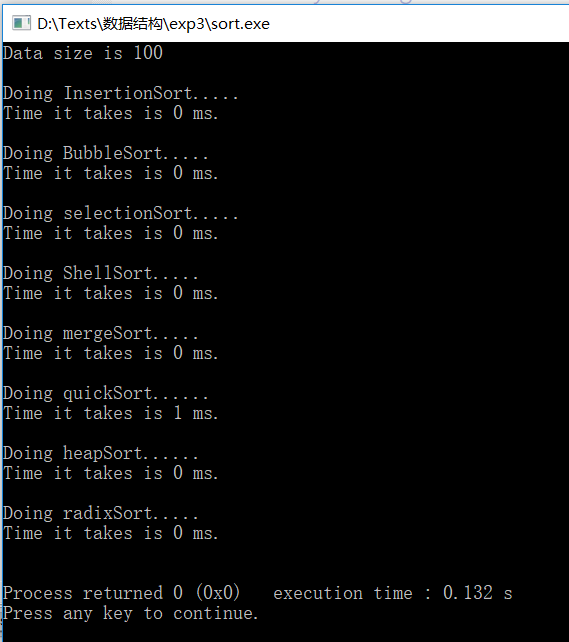
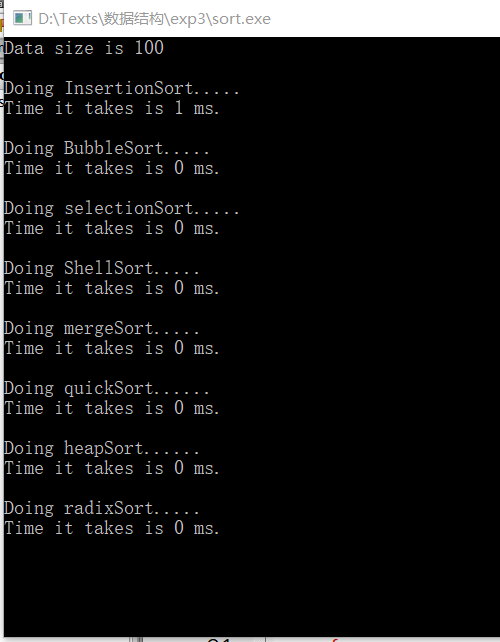
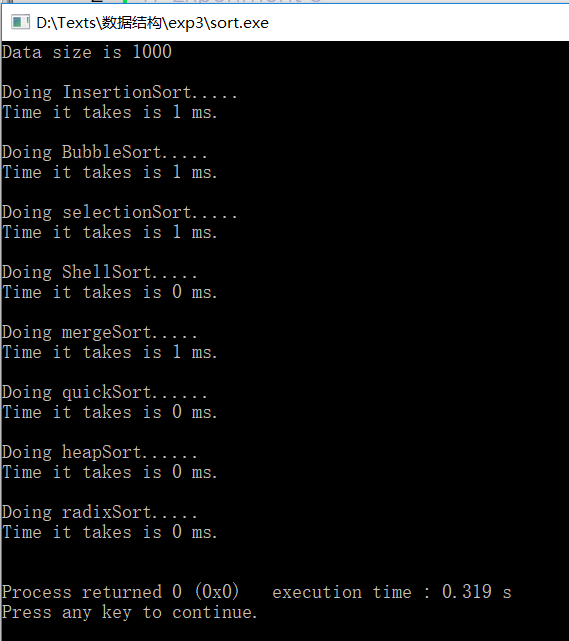
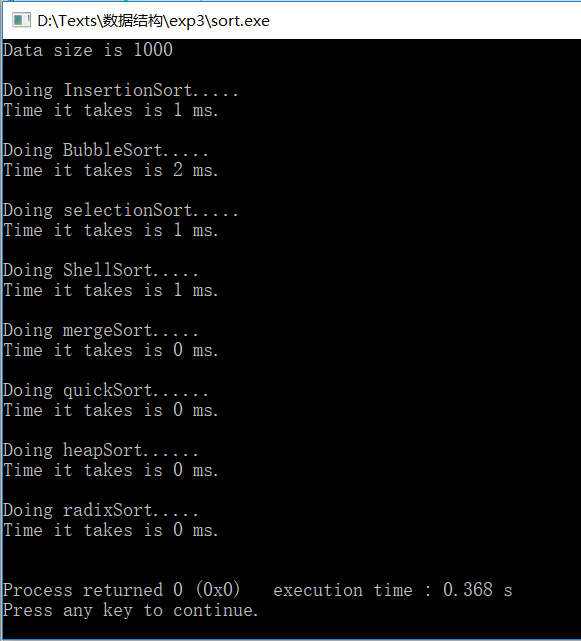
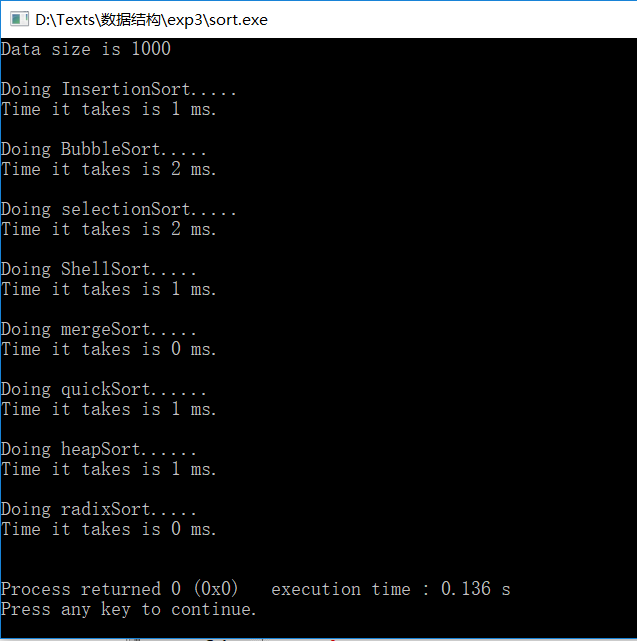
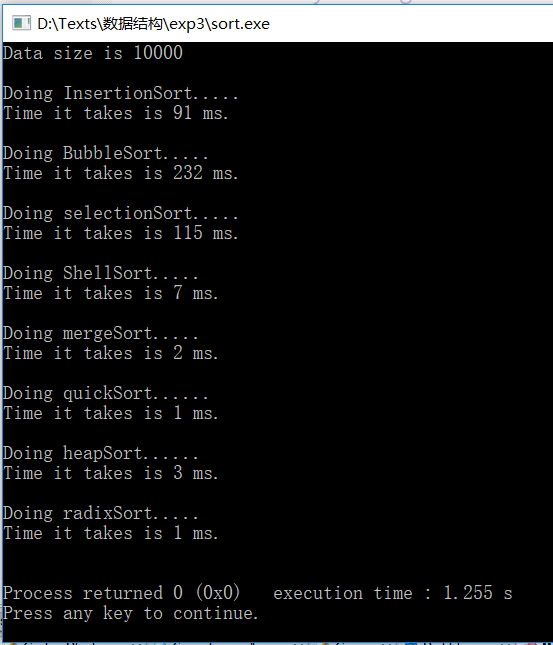
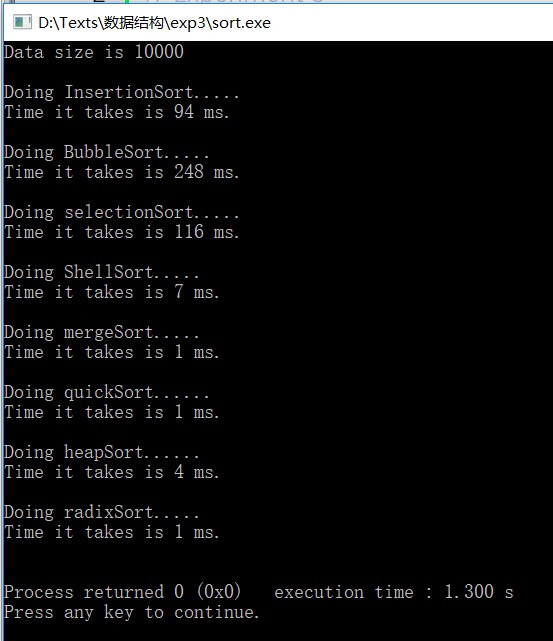
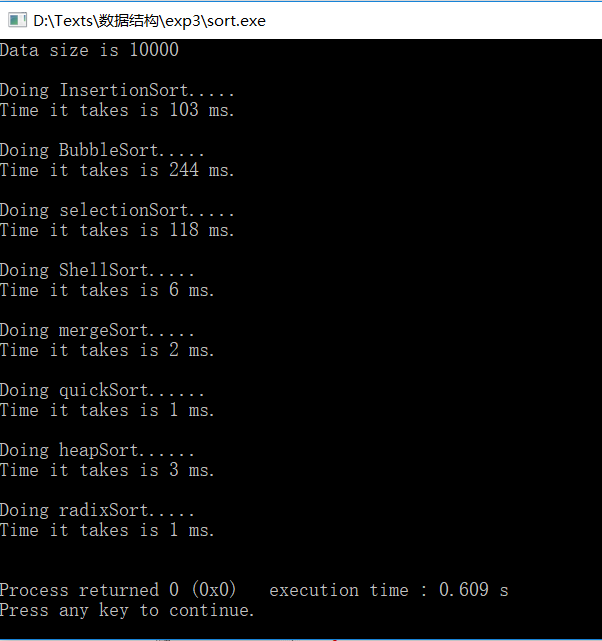
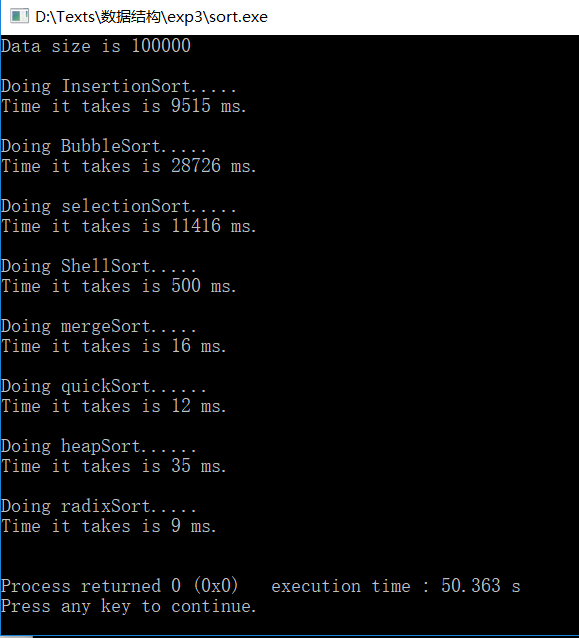
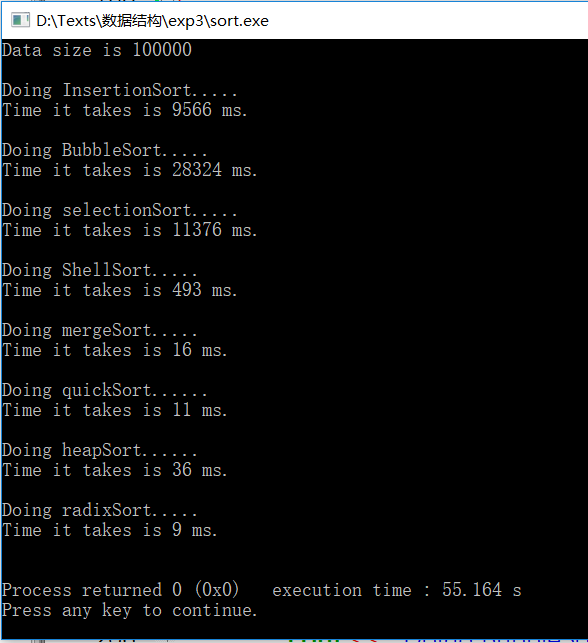
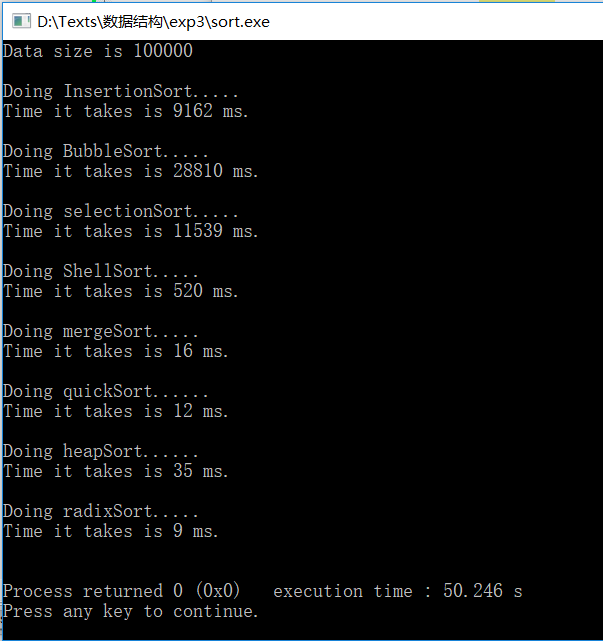
3、内排序算法实现及分析

3.1 实验数据

表1 实验数据

|  |  |  |
| --- | --- | --- |
| 组号 | 规模 | 随机自然数序列 |
| 1 | 100 | 每次运行时随机生成 |
| 2 | 1000 | 每次运行时随机生成 |
| 3 | 10000 | 每次运行时随机生成 |

3.2 程序运行结果



3.3 实验结果表

表2 各算法执行排序的时间开销表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 算法 | 渐进分析  时间复杂度 | 平均执行时间 | | |
| 100 | 1000 | 10000 |
| InsertSort | O(n^2) | 0.33 | 1 | 96 |
| BubbleSort | O(n^2) | 0 | 1.67 | 231.33 |
| SelectionSort | O(n^2) | 0 | 1.33 | 116.33 |
| ShellSort | O(n log n) | 0 | 0.67 | 6.67 |
| MergeSort | O(n log n) | 0 | 0.33 | 1.67 |
| QuickSort | O(n log n) | 0.33 | 0.33 | 1 |
| HeapSort | O(n log n) | 0 | 0.33 | 3.33 |
| RadixSort | O(n log n) | 0 | 0 | 1 |

3.4 实验分析

由折线图、程序运行截图以及前面的表格可以看出，数据规模不大（100与1K）时，所有算法的执行时间都为微秒级（以毫秒为单位输出为0）；当数据规模为10K时，插入排序、选择排序、冒泡排序的执行时间约为其余几种的100倍；当数据规模为100K时，插入排序、选择排序、冒泡排序的执行时间约为其余几种的1000倍；当数据规模为1M时，插入排序、选择排序、冒泡排序的执行时间约为其余几种的10000倍；

5、总结

指数增长的速度是非常快的，而简单的算法带来的几乎是必然的执行效率的低下，这一点在规模小的时候体现不出来，但一旦规模增大，算法的劣势就会明显地表现出来。