**暨南大学本科实验报告**

课程名称 算法分析与设计 成绩评定

实验项目名称 第一周课程实验 指导教师 李军

实验项目编号 01 实验项目类型 设计 实验地点 机房

学生姓名 倪煜麟 学号 2016054314

学院 电气信息学院 专业 软件工程

实验时间 2019年 3月 4 日

**一、实验目的**

* 学习、熟悉各类排序算法；
* 学习分治的思想；
* 测试不同算法的时间复杂度差异；

**二、实验环境**

Dev C++ 5.7.1 编程环境。

**三、实验内容**

1. A.编写一个计时器，它包括至少两个函数start和end。可以把一段程序之前调用start,在程序段的末尾放置end，它能记录在这段程序执行的耗时。
2. 编写一段循环测试之。同段程序多测几次，它稳定吗？

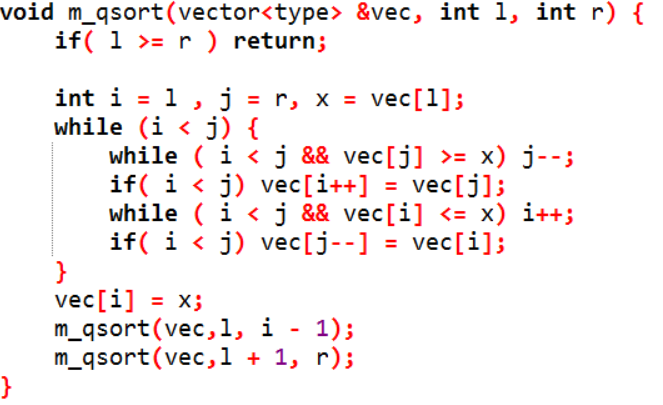
解：实验结果见 Test1Results.txt。

循环20次测试，每次代码如下：



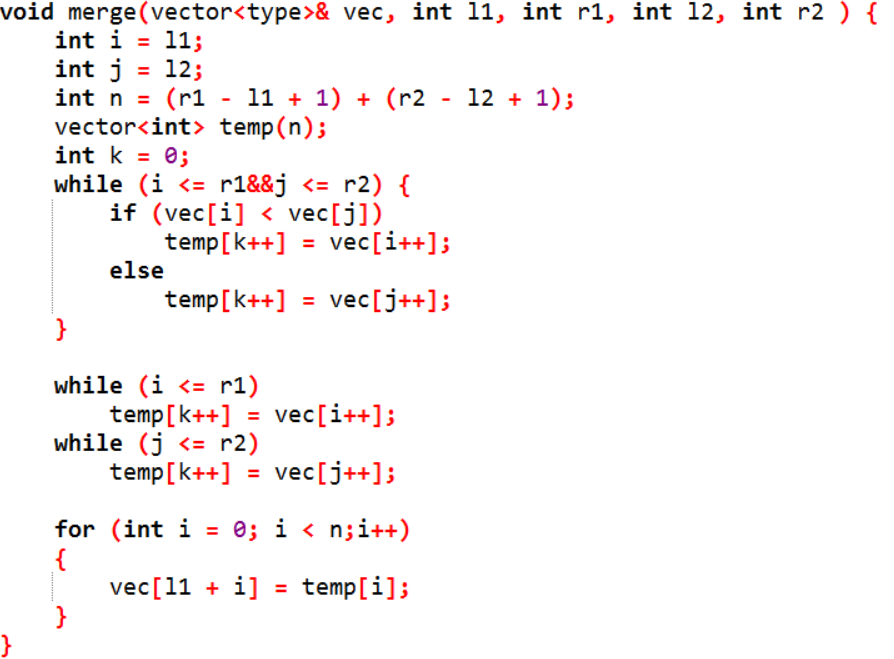
测试时常分布在0.016s和0.024s两点上，误差在可接受范围内，即它是稳定的。

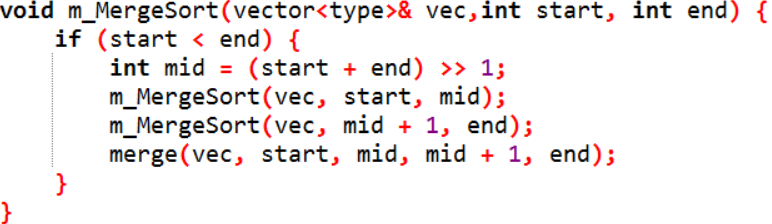
1. 分治法-快速排序（参考3.2.1节）

代码如下：

1. 分治法-归并排序（参考3.2.2节）

代码如下：

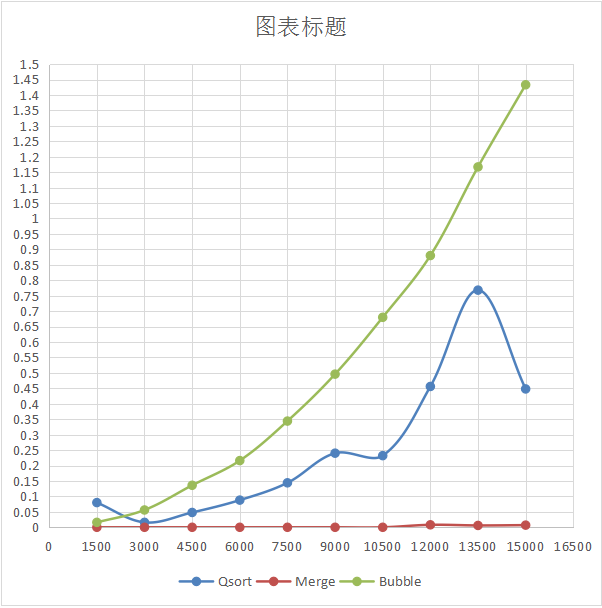




归并排序和快速排序的测试数据、排序结果、时间长分别见Test Merge 和Test Qsort 文件夹

1. 编写一个直接选择排序或冒泡排序，用多组不同长度（n）的数据测试者三者的执行时间，用Excel制作如下的图：

测试10组数据，数据量1500 - 15000，每组差1500个。



可见，归并排序在15000数据量下，运行时间基本可忽略不计，分析其原因，牺牲空间换取时间优势。

数据量超过20000时，c++ STL vector的容量难以支持，故在此仅测试15000数据量。

在此数据范围内，冒泡排序时常呈现出N^2形式增长，符合其规律。

快排则因数据量不够，不能说明其时间复杂度规律。

归并排序则更难看出规律，不过可以通过计算认为，快排和归并排序虽有较大的时间差距，但均符合n\*log(n) 这一数量级。

实验结果见Sort Test文件夹

5.分治法-查找最大和次大元素（参考3.3.1）

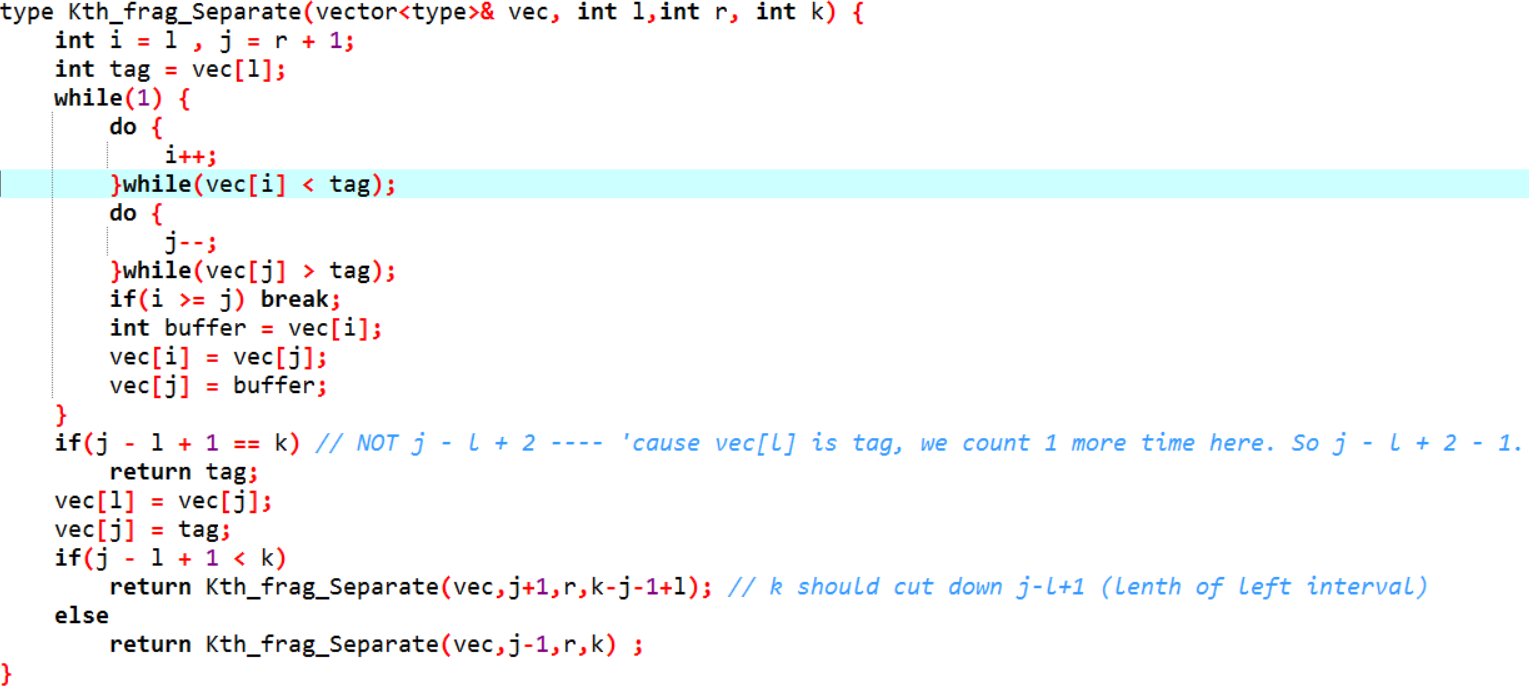
6.编写直接查找的算法，仿照4那样给出实验结果。

7.分治法—寻找一个序列中第k小的元素（参考3.3.3）。同时编写直接算法，给出对比的实验结果。

实验5、7具有同质性，可认为最小元素即为K=1时第K小的元素。

故在此均使用查找第K小元素的子程序。

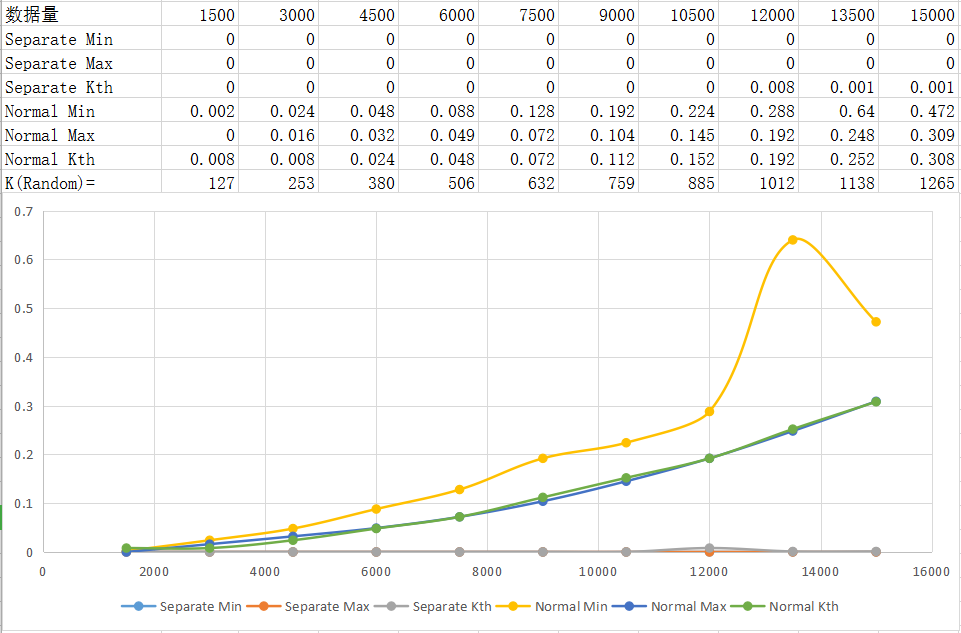
分治查找代码如下（直接查找代码略）：



使用快排思想，使用常数级额外空间。

查找数据及查找时长结果见Find Test文件夹。

查找十组数据，给出对比图如下（数据量1500-15000，区间1500）：



实验所有代码见当前目录，使用Dev C++编译环境编译，头文件使用bits/stdc++.h，不同编译环境可能不支持此头文件，若不能编译请更换头文件。

Random.cpp用于生成随机数据供测试，不属于本次实验范围内代码。

代码部分函数为C++11版本函数，若编译环境不支持，请添加-std=c++11到编译环境参数内。