

# 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

指导教师：金澈清

上机实践名称：线性时间排序

上机实践编号：No.5

年级：21 级

姓名：杨茜雅

学号：10215501435

上机实践成绩：

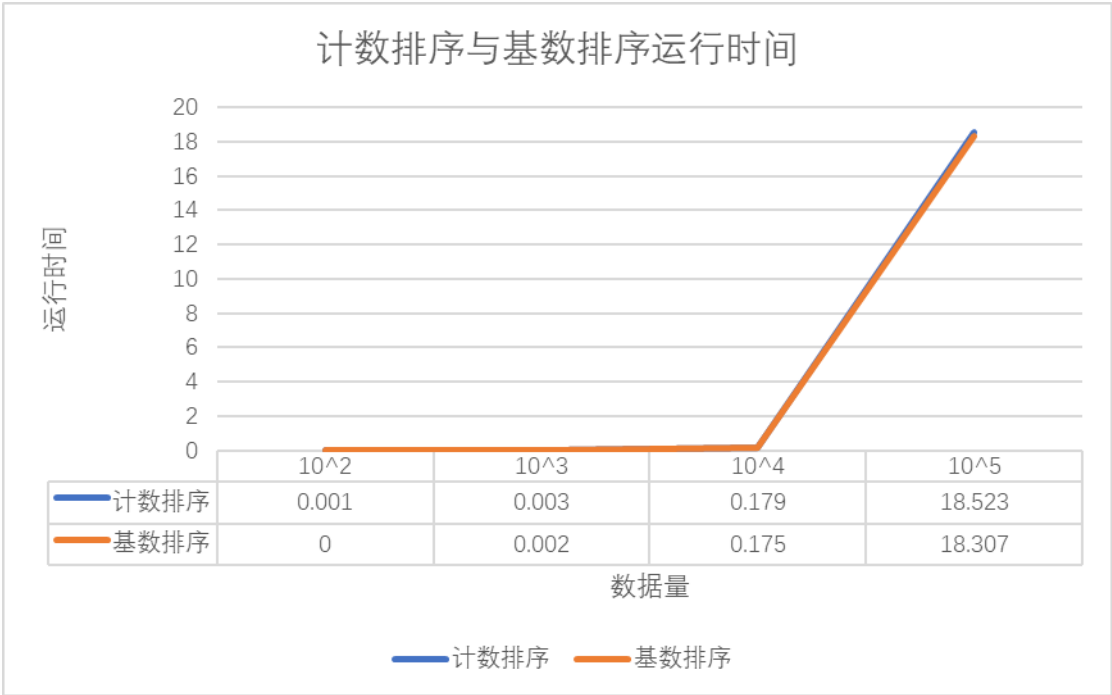
上机实践日期：3.24

## 一、目的

- 1. 熟悉算法设计的基本思想
- 2. 掌握计数排序和基数排序算法

## 二、内容与设计思想

- 1. 完成 OJ 上的题目。
- 2. 用以前写的随机数生成器生成规模为  $10^2, 10^3, 10^4, 10^5$  的随机数据规模，使用 2 种线性时间排序算法进行排序，并画图描述不同数据规模下的 2 种算法的运行时间情况。



## 三、使用环境

推荐使用 C/C++集成编译环境。

## 四、实验过程

- 1. 编写计数排序和基数排序的代码
- 2. 作图展示 2 种算法的运行时间折线图

### 3. 对比线性时间排序算法和以前学过的基于比较的排序算法，探讨二者的适用场景

## 五、总结

插入排序，归并排序，堆排序，快速排序是比较排序，即在排序的最终结果里，元素之间的次序依赖于它们之间的比较。每个数都必须和其他数进行比较，才能确定自己的位置。

计数排序和基数排序是非比较排序，通过确定每个元素之前，应该有多少个元素来排序。针对数组 `array`，计算 `array[i]` 之前有多少个元素，则唯一确定了 `array[i]` 在排序后数组中的位置。

非比较排序只要确定每个元素之前的已有的元素个数即可，所有一次遍历即可解决

插入排序（稳定）：

时间复杂度：

最好时间复杂度： $O(n)$

平均时间复杂度： $O(n^2)$

最坏时间复杂度： $O(n^2)$

空间复杂度： $O(1)$

应用场合：1. 元素特别少

2. 数组基本有序

归并排序（稳定）：

#时间复杂度： $O(n \log n)$ ；

空间复杂度： $O(n)$

堆排序（不稳定）：

时间复杂度： $O(n \log n)$ ；空间复杂度： $O(1)$

适用场合：在  $n$  个元素里找前几个最大的或最小的，我们用堆，并且找大的用小堆，找小的用大堆。还可以模拟优先队列

快速排序（不稳定）：

平均时间复杂度： $O(n \log n)$

#最坏时间复杂度： $O(n^2)$ （有序的情况）

#空间复杂度：需要一个栈空间

计数排序和基数排序都是稳定的排序：

时间复杂度都是  $O(n)$

