

排序算法实验报告

软件 13 杨楠 2021010711

摘要：比较了几种经典的排序算法：插入排序、希尔排序、快速排序、归并排序、基数排序的效果。理论分析了这几种算法的复杂度，并通过代码设计进行验证。程序中随机生成若干数，比较不同数量级下，各算法的运行时间。

1 实验环境

使用 C++ 语言在 VSCode 平台编写代码，在 Windows 环境下编译运行。

2 算法分析

2.1 插入排序

从头遍历整个数组，每次选择这个数，通过与前面的数比较的方式，移动到数组的前面。每次移动的结果是，这个数前面的数都比它小，后面的都比它大。总的时间复杂度是 $\Theta(n^2)$ 。

2.2 希尔排序

将数组划分为等间距的若干部分，相同间距的每一组数进行插入排序。之后将间距逐渐缩小，重复操作，直到最后间距变为 1。由于经过了预排序，到后面实际的比较次数相比于原版的插入排序会减少很多。不同的间距缩小方式对于时间复杂度的影响不同。本实验设置间距 gap 的缩小方式为 $gap = gap / 3 + 1$ ，时间复杂度约为 $O(n^{1.3})$ 。

2.3 快速排序

在数组中选取某个数，以此将数组划分为两部分，比这个数小的放左边，比这个数大的放右边。然后用递归的方法，分别对这两组数重复上述操作。算法的期望运行时间是 $O(n \lg n)$ 。

2.4 归并排序

将数组从中间分成左右两等分，这两部分通过递归进行排序，再合并。合并时，由于这两组数内部已经有序，可以在线性时间内合并完成。从而算法的时间复杂度为 $\Theta(n \lg n)$ 。

2.5 基数排序

对于若干个 b 位数，选定某个基数 r ，这样可以处理为 $d = \lceil \frac{b}{r} \rceil$ 位数。从最低有效位开始，每位进行稳定的排序（通常选择计数排序这种线性排序），循环到高位。每一位的排序所用时间为 $\Theta(n + 2^r)$ ，总的时间复杂度为 $\Theta\left(\frac{b}{r}(n + 2^r)\right)$ ，由教材内容，当 r 选取为 $\lceil \lg n \rceil$ 时，可以得到最优的时间代价 $\Theta\left(\frac{bn}{\lg n}\right)$ 。

3 实验设计思路

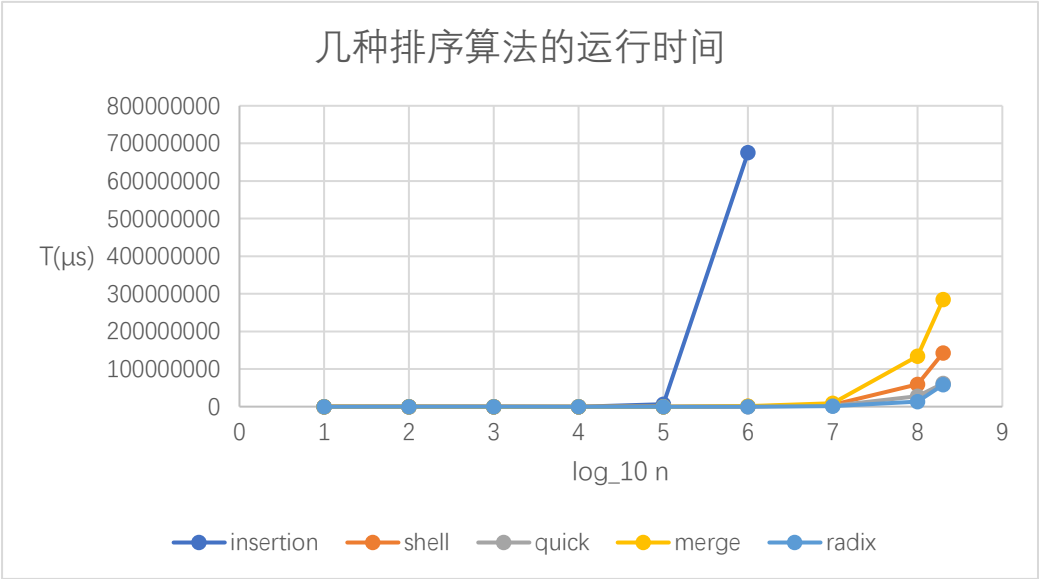
分别在不同的 `cpp` 文件中实现各个算法。运行时，输入相应的 `n` 后，随机生成 `n` 个数进行排序，输出排序所用的时间（单位是微秒）。
生成随机数使用的是`<random>`库，生成的范围是 $[0, 2^{32} - 1]$ 的整数。计算时间使用的是`<windows.h>`库，使用电脑硬件自带的定时器，精度是微秒。

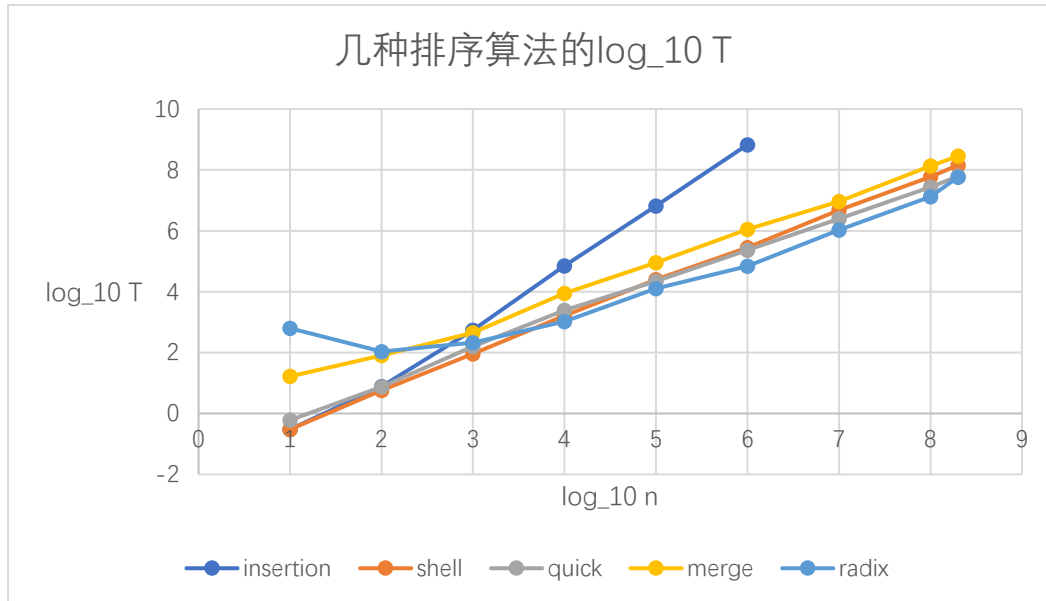
4 结果分析

实验设置的数据量分别为： $10, 10^2, 10^3, 10^4, 10^5, 10^6, 10^7, 10^8, 2 \times 10^8$
整理输出的结果，制成表格如下。

n	10	100	1000	10000	100000	1000000	10000000	100000000	200000000
log ₁₀ n	1	2	3	4	5	6	7	8	8.301029996
insertion	0.3	7.7	550.9	70438.4	6499754	675662763			
shell	0.3	5.7	90.5	1632.2	24995	283890.2	4818566.4	59020359.4	142791550
quick	0.6	7.5	159.9	2453.2	22038.8	229374.4	2505798.9	27422959.5	61589207.3
merge	16.5	80	454	8738.1	89334.8	1131707.9	9107731.4	133993584.9	284590607.1
radix	630.1	108.1	214	1045.4	12828.5	68881.5	1059533.7	13264393.8	58152169.1

分别绘制出 T 与 $\log_{10} n$ 的关系曲线，以及 $\log_{10} T$ 与 $\log_{10} n$ 的关系曲线，如下。





由实验结果可得：

插入排序的时间在 n 为 10^5 的时候就超过了 1 秒，到 10^6 的时候已经是上百秒的级别，可以预见 n 更大时所用的时间将是非常长的。

希尔排序相较于插入排序，有着明显的优势，在时间上也并不比剩下的 3 个算法差多少。

快速排序和归并排序相比，尽管都是 $O(n \lg n)$ 的时间复杂度，但是前者由于是原地排序，地址命中高，所用时间比后者更少些。

基数排序在所选的 r 值最优的情况下，运行时间基本是线性的，在 n 数量级达到 10 的 7 次方，8 次方的时候，所用时间也比其他几种 $O(n \lg n)$ 的算法更少。

5 结论

综合比较以上 5 种排序算法，可以看出，这几种算法各有各的特点。插入排序在数据量比较小的时候具有优势，但不适合数据量较大的情况。希尔排序是插入排序的改进算法，尽管也是多项式时间复杂度，但指数降低了，运行时间降低了不少。快速排序和归并排序适用于数据量较大的情况，但前者更加常用，其在空间复杂度上也具有优势。基数排序的运行是线性时间，但不同的基数设置，性能也会有不同，需要选择最优的基数。

参考文献

CLRS, Introduction to Algorithms (3rd edition), (2009), The MIT Press