

# 排序算法类的实现

陈科辉 Keiver Pabula

24 November, 2022

## 1 设计思路

首先对于 heapsort 和 quicksort 的设计我使用了与书一质的设计思路, 只是我从网上拿了更全的 sort 头文件, 只是调试时发现书顺序需要从新排序, 否则会出现错误。测试的思路首先因为要求分别比较 1%, 10%, 90%, 99% 输入有序的情况下的排序效率, 又因为数组数据不小于 10000, 所以我用  $p=10000 \times$  所要求的数据得到我应该从哪个数开始没有序。对于前面有序的从 1 开始到  $p+1$  是有序的, 而无序的方法我就用逆序从  $n$  到  $p+1$  开始一个一个插入数组, 因为在排序时逆序是最差情况的无序所以可以。然后因为要用多组数据测试, 所以我用外层循环来规定, 然后每次为了不同的数据所以我就用 10000 的  $k$  倍 ( $k=1,2,3,4,5$ ) 来当作数组元素, 对于重复实验我就重复的进行同个实验, 将时间综合, 再除于重复的次数得到平均消耗的时间。

## 2 测试结果

有所设计的测试, 可以得到如下的结果:

```

student@student-VirtualBox:~/ds2022/ds2022/sorting$ ./test 对于n = 10000前1%有序Heapsort花费了 :0.006324s
对于n = 20000前1%有序Heapsort花费了 :0.0111961s
对于n = 30000前1%有序Heapsort花费了 :0.0175988s
对于n = 40000前1%有序Heapsort花费了 :0.0259369s
对于n = 50000前1%有序Heapsort花费了 :0.0339955s
对于n = 10000前1%有序Quicksort花费了 :0.0018634s
对于n = 20000前1%有序Quicksort花费了 :0.0040137s
对于n = 30000前1%有序Quicksort花费了 :0.0072142s
对于n = 40000前1%有序Quicksort花费了 :0.0089224s
对于n = 50000前1%有序Quicksort花费了 :0.0095818s
对于n = 10000前10%有序Heapsort花费了 :0.0054709s
对于n = 20000前10%有序Heapsort花费了 :0.0116802s
对于n = 30000前10%有序Heapsort花费了 :0.0208694s
对于n = 40000前10%有序Heapsort花费了 :0.0265792s
对于n = 50000前10%有序Heapsort花费了 :0.031147s
对于n = 10000前10%有序Quicksort花费了 :0.001656s
对于n = 20000前10%有序Quicksort花费了 :0.0033442s
对于n = 30000前10%有序Quicksort花费了 :0.0051067s
对于n = 40000前10%有序Quicksort花费了 :0.0073074s
对于n = 50000前10%有序Quicksort花费了 :0.009419s
对于n = 10000前90%有序Heapsort花费了 :0.005808s
对于n = 20000前90%有序Heapsort花费了 :0.0120545s
对于n = 30000前90%有序Heapsort花费了 :0.0184096s
对于n = 40000前90%有序Heapsort花费了 :0.0251065s
对于n = 50000前90%有序Heapsort花费了 :0.0336756s
对于n = 10000前90%有序Quicksort花费了 :0.0008899s
对于n = 20000前90%有序Quicksort花费了 :0.0019242s
对于n = 30000前90%有序Quicksort花费了 :0.003167s
对于n = 40000前90%有序Quicksort花费了 :0.0041351s
对于n = 50000前90%有序Quicksort花费了 :0.0055494s
对于n = 10000前99%有序Heapsort花费了 :0.0056517s
对于n = 20000前99%有序Heapsort花费了 :0.0119087s
对于n = 30000前99%有序Heapsort花费了 :0.0184219s
对于n = 40000前99%有序Heapsort花费了 :0.0251087s
对于n = 50000前99%有序Heapsort花费了 :0.03309s
对于n = 10000前99%有序Quicksort花费了 :0.000848s
对于n = 20000前99%有序Quicksort花费了 :0.0019102s
对于n = 30000前99%有序Quicksort花费了 :0.0031782s
对于n = 40000前99%有序Quicksort花费了 :0.0040551s
对于n = 50000前99%有序Quicksort花费了 :0.0055362s

```

不难看出同样的  $n$  下，有序的数组越多不管是 Heapsort 还是 Quicksort 它所消耗的时间都是越短的。当然当  $n$  个数组下，如果数字越小它的排序速度也会越快。最后的在  $n$  一样，有序数组的百分比也一样的情况下，Quicksort 所消耗的时间远远短于 HeapSort。