**《数据结构》实验报告（第7章）**

**学号：\_09020328\_\_\_\_ 姓名 ：\_ 王亮\_\_\_\_**

**实验题号： P435: 4 \_ 实验日期：\_2021.12.6\_\_\_**

**实验一**

**1．问题描述：**

构造出一种复合排序方法，使得其在最坏时间评价下表现良好。

选用范围：(a)插入排序(b)快速排序(c)归并排序(d)堆排序

步骤：

1.完成以上方法，其中快排使用中值方法，归并使用迭代方法，（比较归并的迭代版本与递归版本的运行时间）并且验证正确性；

2.生成四种排序的最坏情况数据

**2．算法思想：**

考虑到以下因素：

1.最坏时间复杂度拟合

2.稳定生成的最坏数据

对以下排序算法的最坏数据生成作特殊处理：

归并：随机数据，使用二进制翻转比较次数应该是最多的，但是时间上比不过随机数据，感觉很奇怪。

堆排序：使用从小到大序列数据，逐次调用堆插入（O(nlogn)）方法而不是一次O(n)建堆。

快排：（不使用中值方法）选取第一位

插排：反序

**3．功能函数：**

Static 用于逐个调试

static void countWorst()：最坏情况时间计算

void testify(int first,int last)：对于给定[first,last)区间进行正确性验证

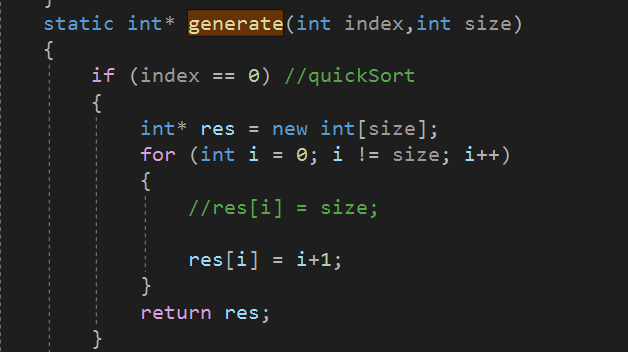
static void countTimeAll()：计算随机情况下所有排序的时间

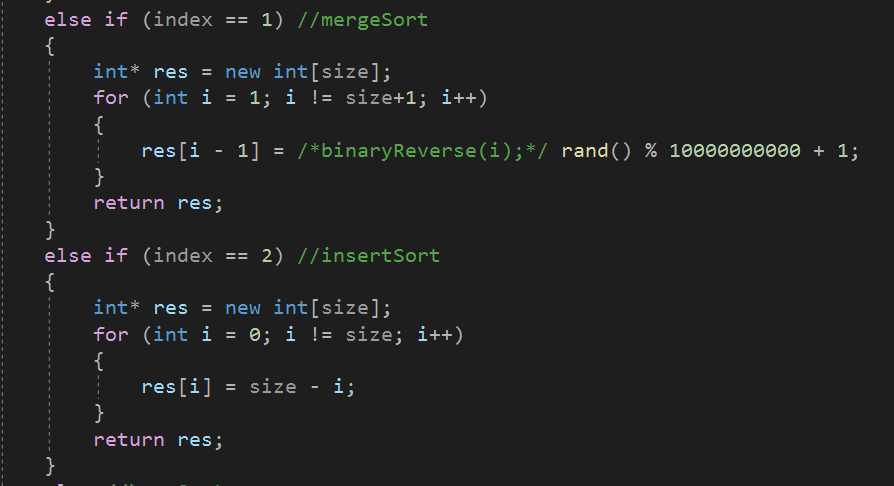
static int\* generate(int index,int size)：生成最坏数据

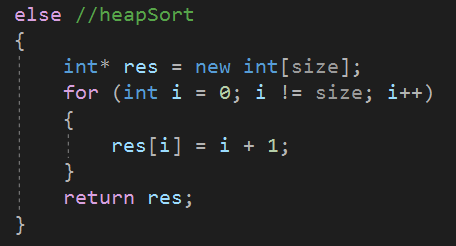
采用clock\_t clock()函数计时 单位：ms

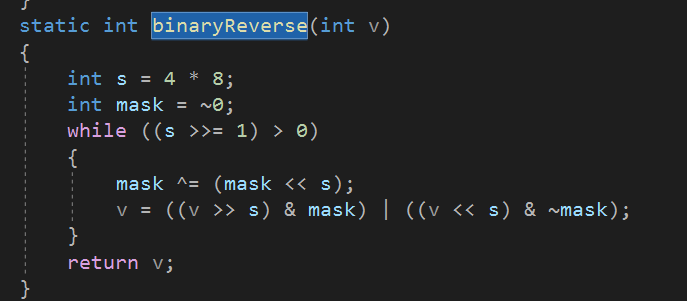
**4．测试数据：**

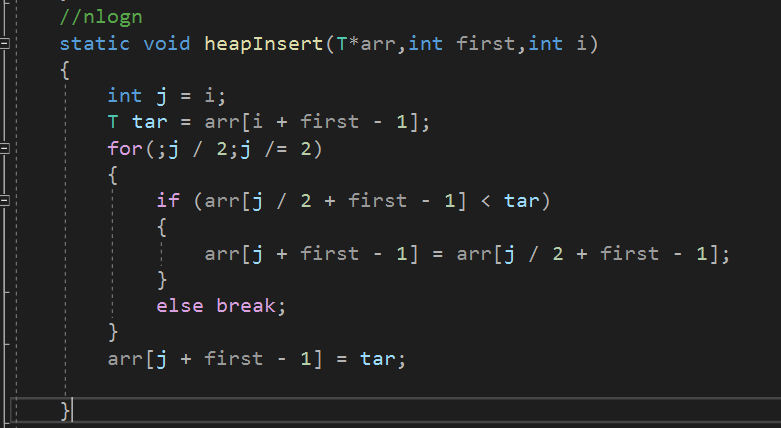
见下方generate()函数：

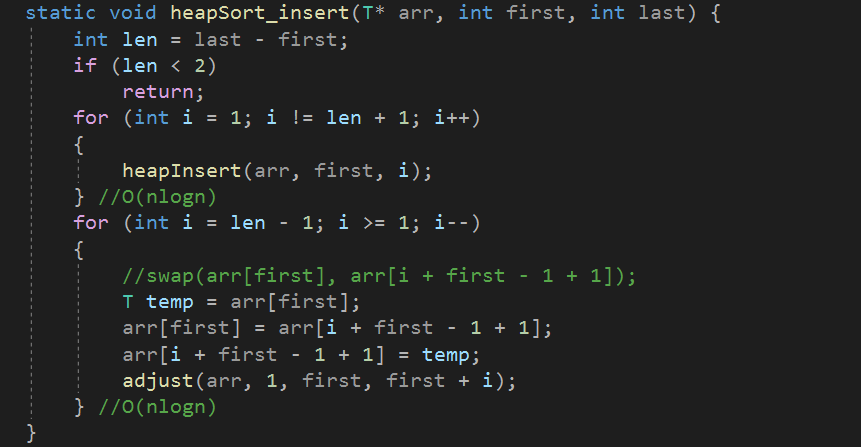




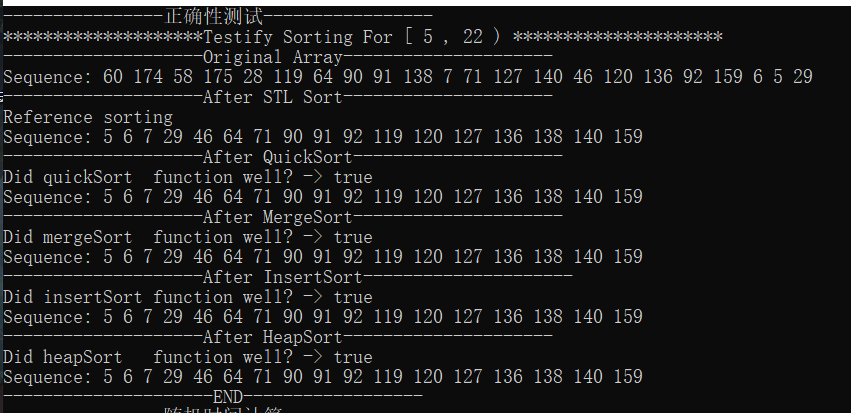


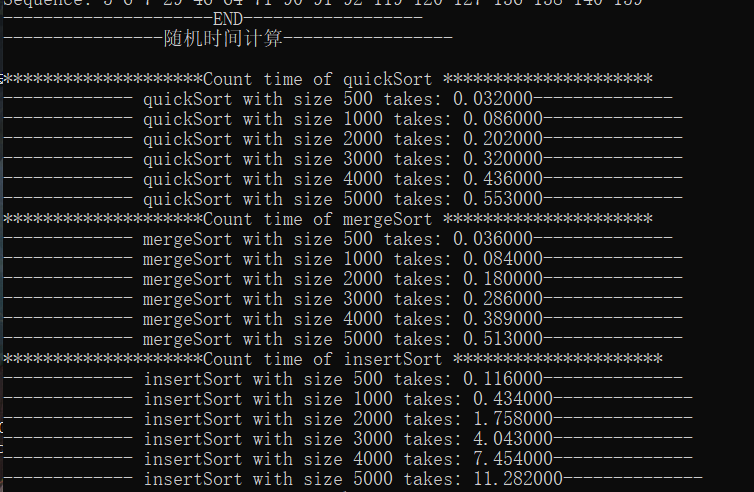


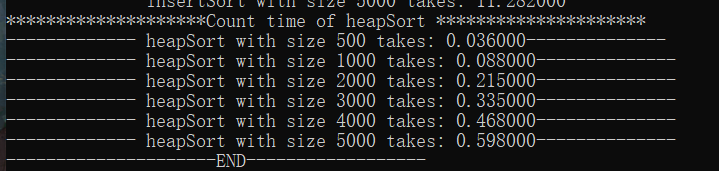


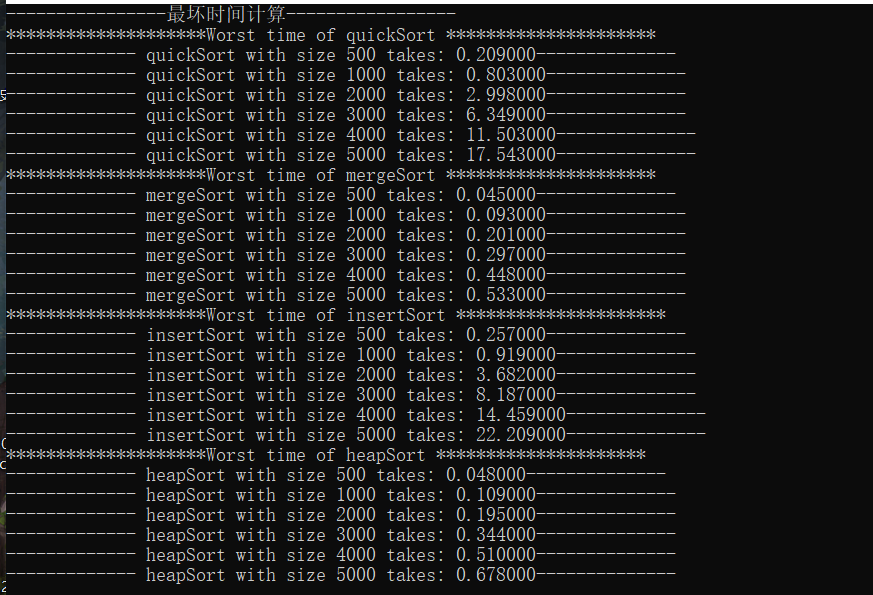


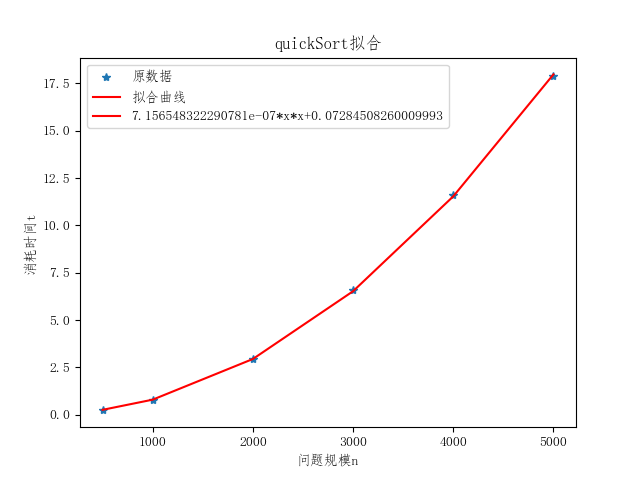
**5．测试情况：**

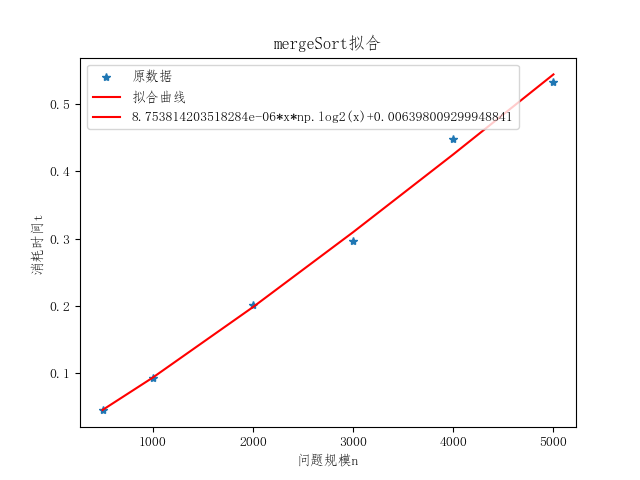


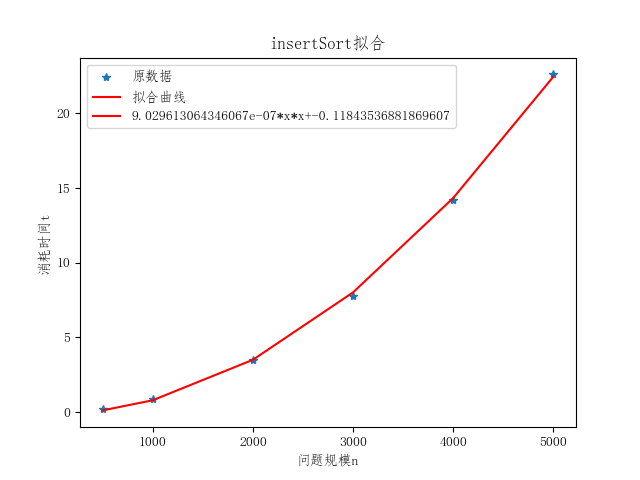


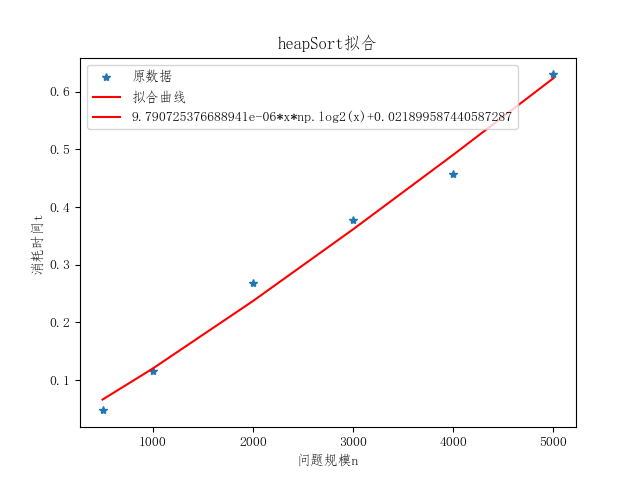


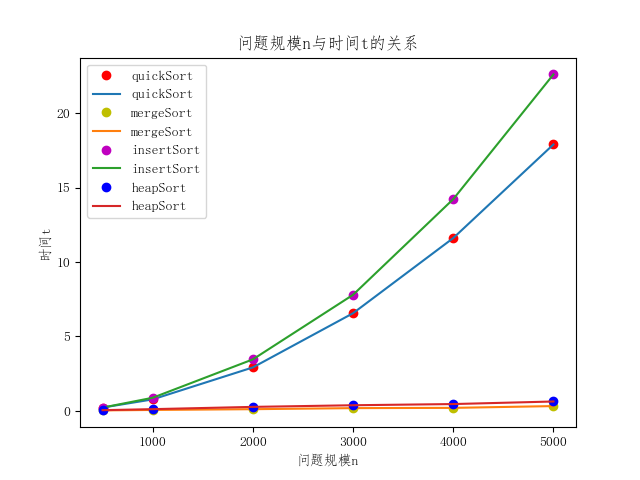




****







**6．实验总结：**

**上述图像拟合较好地验证了各个排序的最坏时间复杂度。如果要用更大规模的数据做效果会更好，但是最坏情况下的快排可能递归过深导致程序崩溃。运行次数较多时这一点十分明显，能确切感受到左循环右递归的快排写法带来的美妙提升。在100 0000 量级及以上 可以测出稳定的快排快于归并排序。最坏情况总体上归并仍是最快，可能在部分区间（完整测试1-5000）出现堆排快于归并，但不影响总体，若以拟合函数作为参考，更为明显。**

**7. 源代码：**

实验源程序清单

(1) As-Lab7.cpp