**第十章 内部排序作业**

一、填空（错一个扣1分）

1.在第一趟排序后，不能确保将数据表中某一个元素放在其最终位

置上的排序算法有\_直插、希尔、归并、基数\_\_。

2. 时间效率与关键字初始序列有关的排序算法有\_\_直插、希尔、冒泡、

快速\_。

3. 最坏时间复杂度是O(nlogn)的排序算法是\_\_堆排、归并\_\_\_\_\_\_\_。

3. 空间复杂度是O(n)的排序算法是\_\_\_归并\_\_\_\_\_\_。

4. \_直插、希尔、归并、基数\_\_排序算法可能出现以下情况：在最后一趟排序算法前，所有关键字均不在其最终的位置上。

5.对n个关键字进行直插排序，最好的关键字比较次数是\_\_n-1\_\_\_\_。

6.要对n个元素进行排序，要求即快又稳定，应采用\_\_归并\_\_\_排序。

7. 以第一个元素为基准进行快速排序的第一次划分，下列四个序列中\_\_D\_\_\_\_\_的元素移动次数最多。

A． 1，2，3，4，5 B. 4，2，3，1，5

C． 5，4，3，2，1 D. 2，3，1，4，5

8. 假设大顶堆(50,20,30,15,17,25,18)。交换后重新调整为大顶堆的结果序列为\_\_\_\_\_30、20、25、15、17、18、50\_\_\_\_\_\_\_\_\_\_\_\_。

9.假设对序列10，30，20，15，5，7，8进行升序排序，排序中间得到以下中间结果，请根据中间结果写出使用的排序算法。

A 8，7，5，10，15，20，30 \_\_\_快排\_\_\_\_\_\_

B. 30，15，20，10，5，7，8 \_\_\_\_堆排\_\_\_\_\_\_\_\_\_

C. 10，30，20，15，5，7，8 \_\_\_\_\_\_直插\_\_\_\_\_\_\_\_

D. 10，30，15，20，5，7，8 \_\_\_\_\_\_归并\_\_\_\_

10. 以下描述是否正确。

A．若待排序序列已基本有序，使用冒泡排序优于快速排序。\_√\_\_\_

B．希尔排序、冒泡、归并、基数排序是稳定的排序算法。\_\_×\_\_\_

C．从1000个数据中选择10个最大的，应使用选择排序。\_\_×\_

(课件P109)

D. 内排序要求数据一定要以顺序方式存储。\_\_\_×\_\_\_

11. 对一组数据（84，47，25，15，21）排序，数据的排列次序在排序的过程中的变化为 （1） 15 47 25 84 21 （3） 15 21 25 84 47 （4） 15 21 25 47 84。 则采用的排序是\_\_选择排序\_\_\_\_。

12. 数据序列{8,10,13,4,6,7,22,2,3}只能是\_\_直插\_\_\_排序算法两趟排序后的结果。

二、计算题（80分，每个算法10分）

已知一组记录为：461、74、653、214、34\*、638、86、465、27、34，给出采用直插排序、希尔排序（gap=n/2,n/4…,1）、冒泡排序、快速排序、简单选择排序、堆排序、归并排序、基排序进行**降**序排序时的前四趟排序结果。如果没有四趟，写出该算法的全部排序过程。

解：

461、74、653、214、34\*、638、86、465、27、34

直插排序

1. 461、74、653、214、34\*、638、86、465、27、34
2. 653、461、74、214、34\*、638、86、465、27、34
3. 653、461、214、74、34\*、638、86、465、27、34
4. 653、461、214、74、34\*、638、86、465、27、34

461、74、653、214、34\*、638、86、465、27、34

希尔排序

1）638、86、653、214、34\*、461、74、465、27、34 //gap=5

2）653、465、638、461、74、214、34\*、86、27、34 //gap=2

3）653、638、465、461、214、、86、74、34\*、34、27 //gap=1

461、74、653、214、34\*、638、86、465、27、34

冒泡排序

1）461、653、214、74、638、86、465、34\*、34、27

2）653、461、214、638、86、465、74、34\*、34、27

3）653、461、638、214、465、86、74、34\*、34、27

4）653、638、461、465、214、86、74、34\*、34、27

461、74、653、214、34\*、638、86、465、27、34

快速排序

1）465、638、653、461、34\*、214、86、74、27、34

2）653、638、465、461、74、214、86、34\*、27、34

3）653、638、464、461、86、214、74、34\*、34、27

4）653、638、464、461、214、86、74、34\*、34、27

461、74、653、214、34\*、638、86、465、27、34

选择排序

1）653、74、461、214、34\*、638、86、465、27、34

2）653、638、461、214、34\*、74、86、465、27、34

3）653、638、465、214、34\*、74、86、461、27、34

4）653、638、465、461、34\*、74、86、214、27、34

461、74、653、214、34\*、638、86、465、27、34

堆排序

建小顶堆：27、34\*、86、74、34、638、653、465、214、461

1)交换：461、34\*、86、74、34、638、653、465、214、27

筛选：34\*、34、86、74、461、638、653、465、214、27

2)交换：214、34、86、74、461、638、653、465、34\*、27

筛选：34、74、86、214、461、638、653、465、34\*、27

3)交换：465、74、86、214、461、638、653、34、34\*、27

筛选：74、214、86、465、461、638、653、34、34\*、27

4)交换：653、214、86、465、461、638、74、34、34\*、27

筛选：86、214、638、465、461、653、74、34、34\*、27

461、74、653、214、34\*、638、86、465、27、34

归并排序

1）461、74、653、214、638、34\*、465、86、34、27

2）653、461、214、74、638、465、86、34\*、34、27

3）653、638、465、461、214、86、74、34\*、34、27

4）653、638、465、461、214、86、74、34\*、34、27

461、74、653、214、34\*、638、86、465、27、34

基数排序

1）个位分配收集

638、27、86、465、74、214、34\*、34、653、461

2）十位分配收集

086、074、465、461、653、638、034\*、034、027、214

3）百位分配收集

653、638、465、461、214、86、74、34\*、34、27