**第10章 内部排序**

1. 直接插入排序在最好的情况下的时间复杂度\_\_\_\_\_\_

A O(n)

B O(nlog2n)

C O(log2n)

D O(n2)

1. 用直接插入排序对下面4个序列递增排序，元素比较次数最少的是\_\_\_\_\_\_\_

A 94,32,40,90,80,46,21,69

B 32,40,21,46,69,94,90,89

C 21,32,46,40,80,69,90,94

D 90,69,80,46,21,32,94,40

1. 对数据序列{15,9,7,8,20,-1,4}进行排序，进行一趟排序后变为{9,15,7,8,20,-1,4},则采用的是\_\_\_\_算法

A 直接选择排序

B 冒泡排序

C 直接插入排序

D 堆排序

1. 以下排序算法中，\_\_\_\_\_\_不能保证每趟排序至少能将一个元素放到其最终的位置上

A 快速排序

B 希尔排序

C 堆排序

D 冒泡排序

1. 在以下排序方法中，关键字比较的次数与记录的初始排列次序无关的是\_\_\_\_\_\_

A Shell排序

B 冒泡排序

C 直接插入排序

D 直接选择排序

1. 给出关键字序列{4,5,1,2,8,6,7,3,10,9}的希尔排序过程,步长依次是5,2,1
2. 设用希尔排序对序列{98,36，-9,0,47,23,1,8,10,7}进行排序，给出步长依次是4、2、1，则排序需要\_\_\_\_\_\_趟，写出第一趟结束后，数序中数据的排序次序为\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
3. 对一组记录(54,38,96,23,15,72,60,45,83)进行Shell排序时,取di+1=└di/2┘,0≤i≤t-1，其中t=└log2n┘,d0==n，dt==1，n为待排序记录的个数，则第二趟排序结束后，前四条记录为\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
4. 对一给定数据{84,47,25,15,21}排序，数据的排列次序在排序过程中的变化为

84　47　25　15　21

15　47　25　84　21

15　21　25　84　47

15　21　25　47　84

则采用的排序方法\_\_\_\_\_\_\_\_

A 直接选择排序

B 快速排序

C 冒泡排序

D 直接插入排序

1. 以下序列不是堆的是\_\_\_\_\_\_\_\_

A {100 85 98 77 80 60 82 40 20 10 66}

B {100 98 85 82 80 77 66 60 40 20 10}

C {10 20 40 60 66 77 80 82 85 98 100}

D {100 85 40 77 80 60 66 98 82 10 20}

1. 有一组数据{15,9,7,8,20,-1,7,4}用堆排序的筛选方法建立的初始堆为\_\_\_\_\_\_\_\_

A {-1 4 8 9 20 7 15 7}

B {-1 7 15 7 4 8 20 9}

C {-1 4 7 8 20 15 7 9}

D 以上都不对

1. 对序列R[1..10]{12,13,11,18,60,15,7,20,25,100},用筛选法建立堆，必须从值为\_\_\_\_\_\_\_\_的关键字开始
2. 在堆排序和快速排序中，若原始记录接近正序或反序，则选用\_\_\_\_\_\_\_，若原始记录无序，则最好选用\_\_\_\_\_\_在直接插入和直接选择排序中，若初始数据基本正序，则选用\_\_\_\_\_\_\_，若初始数据基本反序则选用\_\_\_\_\_\_
3. 快速排序在\_\_\_\_\_\_\_\_情况下最不利于发挥其长处

A 要排序的数据量太大

B 要排序的数据中含有多个相同的值

C 要排序的数据个数为奇数

D 要排序的数据基本有序

1. 在利用快速排序方法对一组记录{54,38,96,23,15,72,60,45,83}进行快速排序时，共需要递归调用的次数为 \_\_\_\_\_ ，其中第二次递归调用是对\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_一组记录进行快速排序
2. 下列排序方法中，辅助空间为O(n)的是\_\_\_\_\_\_\_\_

A 希尔排序

B 冒泡排序

C 堆排序

D 归并排序

1. 已知序列{503 87 512 61 908 170 897 275 653 462},采用二路归并排序对该序列作升序排序时的每一趟结果
2. 在归并排序过程中，需归并的趟数为\_\_\_\_\_\_\_\_
3. 指出堆与二叉排序树的区别
4. 证明对于一个长度为n的任意表进行排序，至少需要进行nlog2n次比较
5. 设计一直接插入算法，假设n条记录，存放在R[0...n-1]中，其中R[i+1,..n-1]为有序区，R[0....i]为无序区，将其关键字与有序区记录(从头开始)进行比较，找到一个刚好大于R[i].key的记录R[j]，将R[i...j-1]记录前移，然后将原R[i]插入到R[j-1]处，要求给出每趟结束后的结果
6. 设计一个双向冒泡排序的算法，即在排序过程交替改变扫描方向
7. 有二个递增序列表，设计一个算法将它们归并成一个有序表(假设每个表中没有重复关键字的元素)，归并时重复关键字的元素只归并一次
8. 编写一个把以A[i]为根的子树调整为堆的过程的算法，即对A[i]进行筛运算，其中已知A[i]的左、右子树均已为堆(假设为大根堆)
9. 下面是在一维数组A[s..t]上进行快速排序的递归算法，s和t的初值应为调用该算法的待排序区间的下限和上限

void quicksort (sorttype A[]，int s，int t)

{ i=s；j=t；x=A[s]

while (i<j) {

while (j>i && A[j].key>=x.key)

\_\_\_\_\_\_；

if (j>i) { \_\_\_\_\_\_；i=i+1； }

while (i<j && A[i].key<=x.key)

\_\_\_\_\_\_；

if (i<j) { \_\_\_\_\_\_；j=j-1 }；

}

A[i]=x；

if (\_\_\_\_\_) quicksort (A，s，i-1)；

if (\_\_\_\_\_) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_；

}

**【作业要求：】1、答案用其它颜色标注**

**2、转换为PDF文件，命名为ex10.pdf，12月12日前网上提交**

**【讲课安排：】1、本次习题由鞠璇在12月13日上课时分析并讲解（要准备PPT）**