

座位号：

杭州电子科技大学学生考试卷（ A ）卷

考试课程	数据结构		考试日期	2015 年 月 日	成 绩	
课程号	A101125D	教师号		任课教师姓名	丁宏, 僧德文, 彭伟民, 陈滨	
考生姓名		学号 (8 位)		年 级		专 业

特别提醒：答案写在答题纸中，并请尽量写在一张纸中。

一. 判断题：（每小题 2 分，共 20 分）

1. 线性表采用链式存储必需占用一片连续的存储空间。

(F)
2. 不论线性表采用顺序存储结构还是链式存储结构，删除值为 X 的结点的时间复杂度均为  $O(n)$ 。

(T)
3. 子串“ABC”在主串“AABCABCD”中的位置为 2 或 5。

(F)
4. 满二叉树必定是完全二叉树。

(T)
5. 设输入序列为 1、2、3，则经过栈的作用后只能得到 1 种确定的输出序列。

(F)
6. 一个 n 维数组可以看成由若干个 n-1 维数组组成的线性表。

(T)
7. 稀疏矩阵的压缩存储可以用一个三元组表来表示稀疏矩阵中的非 0 元素。

(T)
8. 整个工程中，任何一个活动被提前或延迟，将会导致整个工程的提前或延迟。

(F)
9. 堆排序和快速排序中，从平均情况下速度最快的角度来考虑最好选择快速排序。

(T)
10. 分块查找的平均查找长度不仅与索引表的长度有关，而且与块的长度有关。

(T)

二. 选择题：（每小题 2 分，共 20 分）

1. 一个算法的时间复杂度为  $(n^3+n^2\log_2n+14n)/n^2$ ，其数量级表示为（ D ）。

(A) $n^3$  (B) $n^2$  (C) $\log_2n$  (D) $n$
2. 栈和队列的共同特点是（ A ）。

(A)只允许在端点处插入和删除元素 (B)都是先进后出

(C)都是先进先出 (D)没有共同点
3. 设有向无环图 G 中的有向边集合  $E=\{<1, 2>, <2, 3>, <3, 4>, <1, 4>\}$ ，则下列属于该有向图 G 的一种拓扑排序序列的是（ A ）。

(A) 1, 2, 3, 4 (B) 2, 3, 4, 1 (C) 1, 4, 2, 3 (D) 1, 2, 4, 3
4. 设有一个二维数组  $A[m][n]$ ，假设  $A[0][0]$  存放在 644(10)， $A[2][2]$  存放在 676(10)，每个元素占一个空间，问  $A[3][3]$  (10) 存放在（ C ）位置。（脚注(10)表示用 10 进制表示）。

(A)688 (B)678 (C)692 (D)696
5. 设顺序线性表中有 n 个数据元素，则删除表中第 i 个元素需要移动（ A ）个元素。

(A)  $n-i$  (B)  $n+1-i$  (C)  $n-1-i$  (D)  $i$
6. 设按照从上到下、从左到右的顺序从 1 开始对完全二叉树进行顺序编号，则编号为 i 结点的左孩子结点的编号为（ B ）。

(A)  $2i+1$  (B)  $2i$  (C)  $i/2$  (D)  $2i-12$
7. 设 F 是由 T1、T2 和 T3 三棵树组成的森林，与 F 对应的二叉树为 B，T1、T2 和 T3 的结点数分别为 N1、N2 和 N3，则二叉树 B 的根结点的左子树的结点数为（ A ）。

(A)  $N1-1$  (B)  $N2-1$  (C)  $N2+N3$  (D)  $N1+N3$

8. 设某有向图的邻接表中有 n 个表头结点和 m 个表结点，则该图中有（ C ）条有向边。

(A)  $n$  (B)  $n-1$  (C)  $m$  (D)  $m-1$
9. 设有序顺序表中有 n 个数据元素，则利用二分查找法查找数据元素 X 的最多比较次数不超过(A )。

(A)  $\log_2n+1$  (B)  $\log_2n-1$  (C)  $\log_2n$  (D)  $\log_2(n+1)$
10. 设一组初始记录关键字为(72，73，71，23，94，16，5)，则以记录关键字 72 为基准的一趟快速排序结果为（ D ）。

(A) 71，73，5，16，72，23，94 (B) 71，73，5，16，23，72，94

(C) 5，16，71，23，72，94，73 (D) 5，16，71，23，72，94，73

三. 填空题：（每空 2 分，共 20 分）

1. 设一棵二叉树的中序遍历序列为 BDCA，前遍历序列为 CBDA，则这棵二叉树的后序序列为(DBAC )。
2. 设一维数组中有 n 个数组元素，则读取第 i 个数组元素的平均时间复杂度为（ $O(1)$ ）。
3. 深度为 k 的完全二叉树中最少有（ $2^{k-1}$ ）个结点。
4. 设某无向图中有 n 个顶点 e 条边，则建立该图邻接表的时间复杂度为（ $O(n+e)$ ）。
5. 广义表  $A(( ), a, (b), c)$ ， $head(head(tail(tail(A))))$ 等于（ b ）。
6. 设某棵三叉树中有 40 个结点，则该三叉树的最小高度为（ 4 ）。
7. 设二叉排序树中有 n 个结点，则在二叉排序树的平均查找长度为（  $O(\log_2n)$  ）。
8. （中）序遍历二叉排序树可以得到一个有序的序列。
9. 设二叉排序树的高度为 n，则在该树中查找关键字 key 最多需要比较（ n ）次。
10. 利用直接插入排序法的思想建立一个有序线性表的时间复杂度为（ $O(n^2)$ ）。

四. 结构问答题：（每小题 6 分，共 30 分）

1. 设在某通信系统中使用了 5 个字符，它们出现的次数分别为 10，20，50，15，5，试构造一棵赫夫曼树，并给出赫夫曼编码。（左 0 右 1，左小右大）
2. 设一组有序的记录关键字序列为(13，18，24，35，47，50，62，83，90)，查找方法用二分查找，要求计算出查找关键字 62 时的比较次数，给出 head，tail，mid 的位置变化，并计算出查找成功时的平均查找长度。
3. 记录的关键码序列为：63，90，70，55，67，42，98，83，10，45，58，要求构造一棵二叉排序树并给出构造过程。
4. 已知一个图的顶点集 V 和边集 E 分别为：  $V=\{1, 2, 3, 4, 5, 6, 7\}$ ；  
 $E=\{(1, 2)3, (1, 3)5, (1, 4)8, (2, 5)10, (2, 3)6, (3, 4)15,$   
 $(3, 5)12, (3, 6)9, (4, 6)4, (4, 7)20, (5, 6)18, (6, 7)25\}$ ；

用克鲁斯卡尔算法得到最小生成树，试写出在最小生成树中依次得到的各条边。
5. 设有向图 G 的二元组形式表示为  $G=(D, R)$ ， $D=\{1, 2, 3, 4, 5\}$ ， $R=\{r\}$ ， $r=\{<1,2>, <2,4>, <4,5>, <1,3>, <3,2>, <3,5>\}$ 。

(1) 画出该图和它的邻接矩阵存储结构。

(2) 画出该图的邻接表存储结构。

(3) 给出该图的一种拓扑排序序列。

五. 算法设计：（10 分）

在顺序存储结构上设计冒泡排序算法。写出算法思路，并给出 C 或 C++实现。

座位号:

杭州电子科技大学学生考试卷（ A ）卷评分标准[14-15-1]

一、 判断题（每题 2 分，共 20 分）

1.F	2.T	3.F	4.T	5.F	6. T	7.T	8.F	9.T	10.T
-----	-----	-----	-----	-----	------	-----	-----	-----	------

二、 选择题（每题 2 分，共 20 分）

1.D	2.A	3.A	4.C	5.A	6. B	7.A	8.C	9.A	10.CD
-----	-----	-----	-----	-----	------	-----	-----	-----	-------

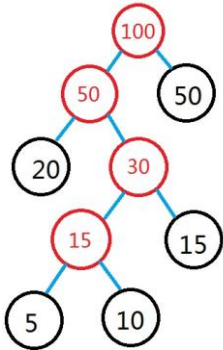
三、 填空题（每空 2 分，共 20 分）

1. DBAC	2. O(1)	3. $2^{k-1}$	4. O(n+e)	5. b
6. 4	7. $O(\log_2 n)$	8. 中	9. n	10. $O(n^2)$

四、 问答题（每题 6 分，共 30 分）

1. (由于有相等的情况，因答案不唯一)

1, 00, 011, 0101,0100



2.

13(head), 18, 24, 35, 47(mid), 50, 62, 83, 90(tail)

13, 18, 24, 35, 47, 50(head), 62(mid), 83, 90(tail)

2 次

, ASL=(1\*1+2\*2+3\*4+4\*2)/9=25/9

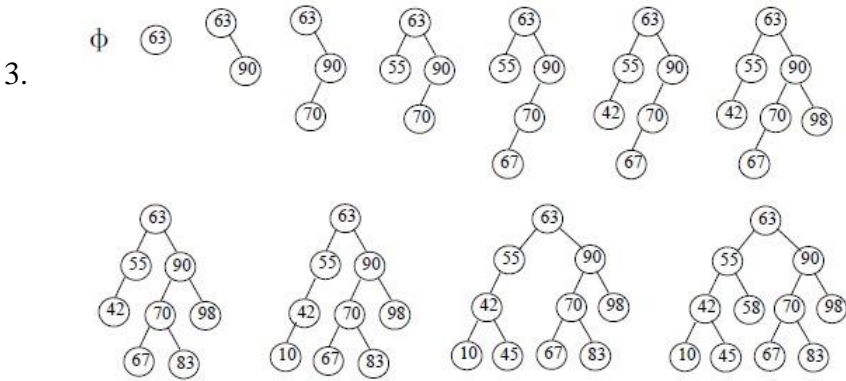


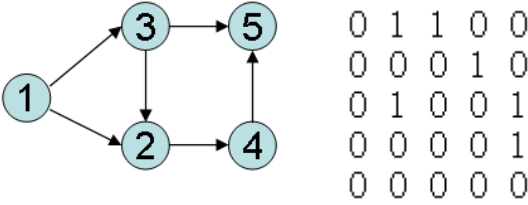
图 9.5 从空树开始建立二叉排序树的过程

4.

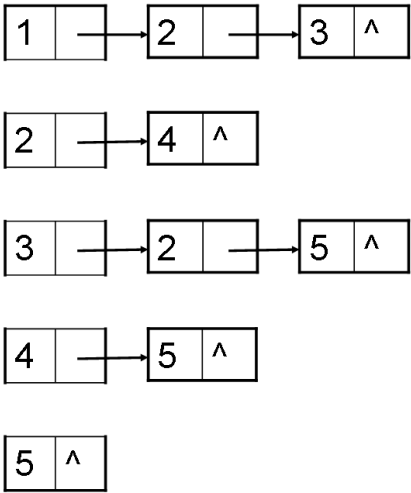
用克鲁斯卡尔算法得到的最小生成树为:

(1,2)3, (4,6)4, (1,3)5, (1,4)8, (2,5)10, (4,7)20

5.



0	1	1	0	0
0	0	0	1	0
0	1	0	0	1
0	0	0	0	1
0	0	0	0	0



一种拓扑排序序列: 1, 3, 2, 4, 5

五、 算法设计题（10 分）

```
void bubble(int r[n])
{
    for(i=1;i<=n-1; i++)
    {
        for(exchange=0,j=0; j< n-i;j++)
        {
            if (r[j]>r[j+1]){
                temp=r[j+1];
                r[j+1]=r[j];
                r[j]=temp;
                exchange=1;}
        }
        if (exchange==0) return;
    }
}
```