

2003 年数据结构期末考题

一. 填空题：(每空 1 分，共 11 分)

1. 设只包含根结点的二叉树的高度为 0, 则高度为 k 的二叉树的最大结点数为_____, 最小结点数为_____。
2. 某二叉树结点的中序遍历序列为 A, B, C, D, E, F, G , 后序遍历序列为 B, D, C, A, F, G, E , 则该二叉树结点的前序遍历序列为_____, 该二叉树对应的树林包括_____棵树。
3. 设有关键码序列 $(Q, H, C, Y, Q, A, M, S, R, D, F, X)$, 要按照关键码值递增的次序进行排序, 若采用初始步长为 4 的 Shell(希尔)排序法, 则一趟扫描的结果是_____ ;若采用以第一个元素为分界元素的快速排序法, 则一趟扫描的结果是_____。
4. 对于顺序存储的栈, 因为栈的空间是有限的, 在进行_____操作时, 可能发生栈的上溢, 在进行_____操作时, 可能发生栈的下溢。
5. 用起泡法对 n 个关键码排序, 在最好情况下, 只需做_____次比较和_____次移动; 在最坏的情况下要做_____次比较。

二. 判断题(下列各题, 你认为正确的, 请在题干的括号内打“√”, 错的打“×”。)(每题 1 分, 共 10 分)

1. 数据结构概念包括数据之间的逻辑结构, 数据在计算机中的存储方式和数据的运算三个方面 ()
2. 线性的数据结构可以顺序存储, 也可以链接存储。非线性的数据结构只能链接存储。()
3. 栈和队列逻辑上都是线性表。 ()
4. 单链表从任何一个结点出发, 都能访问到所有结点。()
5. 将一棵树转换成二叉树后, 根结点没有左子树。()
6. 哈夫曼树是带权路径长度最短的树, 路径上权值较大的结点离根较近。()
7. 用邻接矩阵法存储一个图所需的存储单元数目与图的边数有关。()
8. 在哈夫曼编码中, 当两个字符出现的频率相同时, 其编码也相同。()
9. 线性表采用顺序存储表示时, 必须占用一片连续的存储单元。()
10. 快速排序是一种稳定的排序方法。()

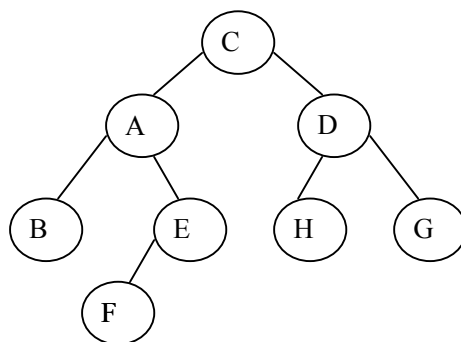
三. 单项选择题, 从每小题后给出的答案中选择一个正确的答案填入括号内。(每题 1 分, 共 9 分)

1. 若长度为 n 的线性表采用顺序存储结构, 在其第 i 个位置插入一个新元素的算法的时间复杂度为()。($1 \leq i \leq n+1$)

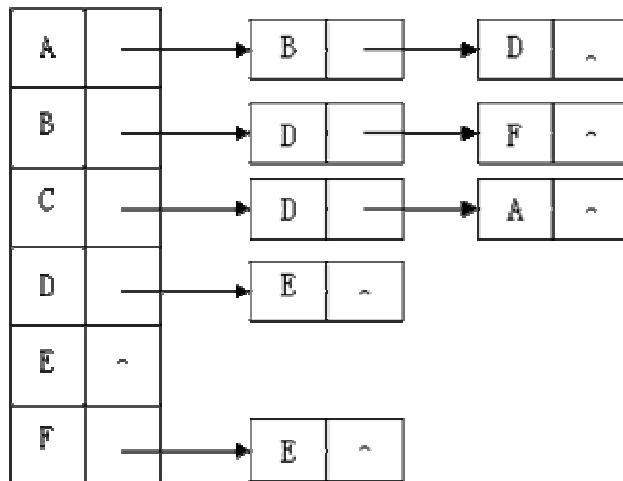
- A . $O(0)$ B . $O(1)$ C . $O(n)$ D . $O(n^2)$
2. 若在线性表中采用折半查找法查找元素, 该线性表应该()。
- A . 元素按值有序 B . 采用顺序存储结构
C . 元素按值有序, 且采用顺序存储结构 D . 元素按值有序, 且采用链式存储结构
3. 具有 n 个顶点的有向图最多有()条弧。
- A . n B . $n(n-1)$ C . $n(n+1)$ D . n^2
4. 从未排序序列中依次取出一个元素与已排序序列中的元素依次进行比较, 然后将其放在已排序序列的合适位置, 该排序方法称为()排序法。
- A . 插入 B . 选择 C . shell D . 二路归并
5. 排序趟数与序列的原始状态有关的排序方法是()排序法。
- A . 插入 B . 选择 C . 冒泡 D . 快速
6. 线性链表不具有的特点是()。
- A . 随机访问 B . 不必事先估计所需存储空间大小
C . 插入与删除时不必移动元素 D . 所需空间与线性表长度成正比
7. 在一个无向图中, 所有顶点的度数之和等于所有边数的()倍。
- A . 3 B . 2 C . 1 D . $1/2$
8. 对有 14 个数据元素的有序表 $R[14]$ 进行折半查找, 搜索到 $R[3]$ 的关键码等于给定值, 此时元素比较顺序依次为()。
- A . $R[0], R[1], R[2], R[3]$ B . $R[0], R[13], R[2], R[3]$
C . $R[6], R[2], R[4], R[3]$ D . $R[6], R[4], R[2], R[3]$
9. 任一棵二叉树, 其叶子结点数为 n_0 , 度为 2 的结点数为 n_2 , 则存在关系()。
- A . $n_2+1=n_0$ B . $n_0+1=n_2$ C . $2n_2+1=n_0$ D . $n_2=2n_0+1$

四 . 综合题

1. (5 分)给出下列二叉树的前序、中序、后序和层次遍历序列。



2. (10 分)下图是用邻接表存储的图, 画出此图, 并根据邻接表写出从 C 点开始按深度优先遍历和广度优先遍历该图的结果。



3. (10 分)已知某系统在通信联络中只可能出现八种字符，其概率分别为 0.07(A)、0.19(B)、0.02(C)、0.06(D)、0.32(E)、0.03(F)、0.21(G)、0.10(H)，
要求：

- (1) 画出哈夫曼树。
- (2) 设计哈夫曼编码。
- (3) 求该树的带权路径长度。

4. (10 分)在如下所示的折半插入排序算法中，添加适当的语句，使之功能完整。
 其中：待排序数据类型为 int，全部数据存储在 Vector 数组中，下标从 0 开始存储，共 n 个。
 提示：折半插入排序算法使用折半查找算法为待插数据确定插入位置，然后使用直接插入算法插入数据

```
Sort()
{
    int left, right, middle, temp;
    // left 表示数组中有序部分的左端下标，right 表示数组中有序部分的右端下标
    for ( int i = 1; i < n ; i++)
        left = 0; right = i-1;
        temp = Vector[i];
        while ( _____ )
        {
            int middle = (left + right) / 2;
            if (temp < Vector[middle])
                _____;
            else
                _____;
        }
        for ( int k = _____; k >= _____; k --) //进行数据移动
            _____;
            _____;
    }
```

5. (15 分)已知 L 是带表头结点的非空单链表，且 P 结点既不是首元结点，也不是尾元结点，试从下列提供的答案中选择合适的语句序列

- 删除 P 结点的直接后继结点的语句序列是 _____
- 删除 P 结点的直接前驱结点的语句序列是 _____
- 删除 P 结点的语句序列是 _____
- 删除首结点的语句序列是 _____
- 删除尾结点的语句序列是 _____

(1) P = P->next	(6) P->next = P	(10) while(P->next->next != NULL) P = P->next;
(2) P = L	(7) P = P->next->next	(11) while(P != NULL) P = P->next;
(3) L = L->next	(8) P->next = P->next->next	(12) while(Q->next != NULL) { P = Q ; Q = Q->next;}
(4) Q = P	(9) free(Q)	(13) while(Q->next != Q) P = P->next;
(5) Q = P->next		(14) while(P->next->next != Q) P = P->next;

6. (10 分)判断以下序列是否为**大根堆**？若否，则按照教材中的算法将它们调整为**大根堆**，要求：画出调整后的堆结构图和相对应的序列(不要求过程)

1.(38,56,25,23,40,100,29,61,35,76,28,20)

2.(21,66,39,73,86,48,52,90,75,88)

3.(12,70,33,65,24,56,48,92,86,33,21)

7. (10 分)在包含 n 个关键码的线性表里进行顺序查找，若查找第 i 个关键码的概率为 P_i ， P_i 如下分布： $P_1=1/2$ 、 $P_2=1/4$ 、... $P_{n-1}=1/2^{n-1}$ 、 $P_n=1/2^n$ 。求成功检索的平均比较次数。

目录

- 1.1 填空题 1
- 1.2 判断题 1
- 1.3 选择题 2
- 1.4 综合题 2

-1.1 填空题

- 1. $2^{k+1} - 1, k + 1$
- 2. $E, A, C, B, D, G, F, 2$
- 3. $QACSQDFXRHMY, FHCDQAMQRSYX$
- 4. 入栈，出栈
- 5. $n-1, 0, n(n-1)/2$

-1.2 判断题

- 1. 错
- 2. 错
- 3. 对
- 4. 错
- 5. 错
- 6. 对

7. 错
8. 错
9. 对
10. 错

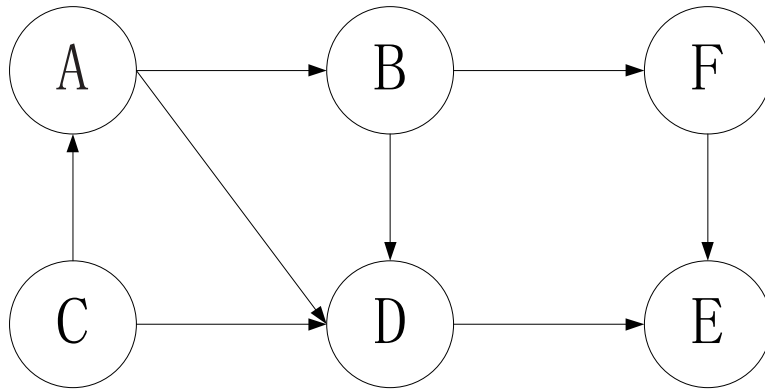
-1.3 选择题

1. C
2. C
3. B
4. A
5. C和D都算对
6. A
7. B
8. C
9. A

-1.4 综合题

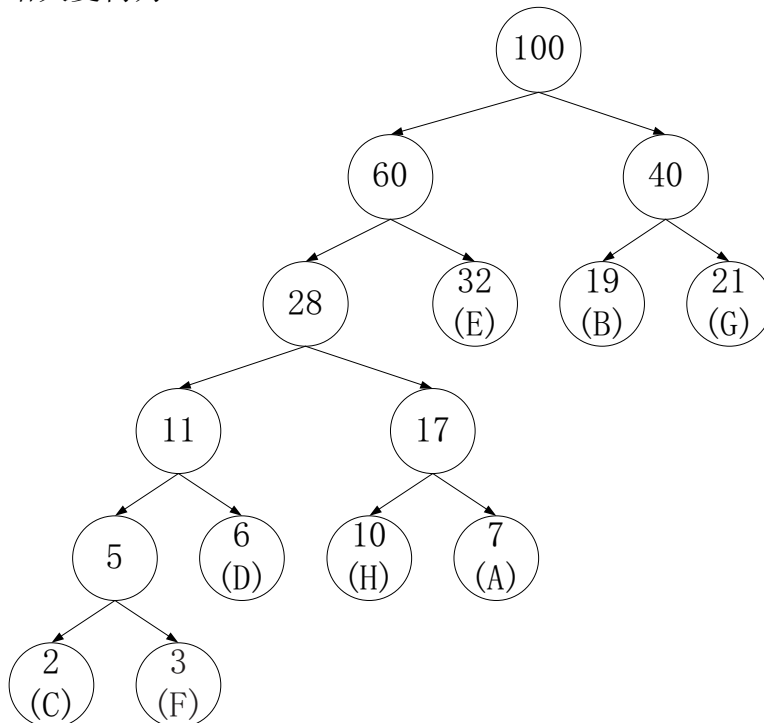
1. 前序 CABEFDHG
中序 BAFECHDG
后序 BFEAHGDC
层次 CADBEHGF

2. 图为



从C点开始深度优先遍历得到的序列为CDEABF（另外一种序列CABDEF也算对），广度优先遍历得到的序列为CDAEBF（另外一种序列CADBEF也算对）

3. 哈夫曼树为




哈夫曼编码为（假设为左0右1）

A 0011

- B 10
- C 00000
- D 0001
- E 01
- F 00001
- G 11
- H 0010

带权路径长度为2.61

4.
 - $left \leq right$
 - $right = middle - 1$
 - $left = middle + 1$
 - $i, middle$ 
 - $Vector[k] = Vector[k - 1]$
 - $Vector[middle] = temp$
5.
 - (a) 5, 8, 9
 - (b) 4,2,14,5,8,9
 - (c) (4,2,14,1,8,9) (5,2,14,5,8,9)
 - (d) (2,5,8,9) (2,4,3,9)
 - (e) (10,5,8,9) (4,12,8,9) (2,4,12,8,9) (2,10,5,8,9)
6.
 - (a) 否, 100 76 38 61 56 25 29 23 35 40 28 20
 - (b) 否, 90 88 52 75 86 48 39 73 66 21
 - (c) 否, 92 86 56 70 33 33 48 65 12 24 21
- 7.

$$ASL_{suc} = S_n = \sum_{i=0}^n \frac{1}{2^i} * i = 1 * 1/2 + 2 * 1/4 + 3 * 1/8 + \dots + n * 1/2^n \quad (1)$$

$$\frac{1}{2} * S_n = 1 * 1/4 + 2 * 1/8 + \dots + n * 1/2^{n+1} \quad (2)$$

(1) – (2)有

$$\frac{1}{2} * S_n = 1 * 1/2 + 1 * 1/4 + 1 * 1/8 + \cdots + 1 * 1/2^n - n * 1/2^{n+1} \quad (3)$$

即

$$ASL_{suc} = S_n = 2 - \frac{1}{2^{n-1}} - \frac{n}{2^n} \quad (4)$$

注：若用 $ASL_{suc} = \frac{\sum_{i=1}^n P_i * C_i}{\sum P_i}$ 也算对