



暨南大学  
JINAN UNIVERSITY

2011 年全国硕士研究生统一入学考试自命题试题

\*\*\*\*\*

学科与专业名称：计算机技术，软件工程

考试科目代码与名称：数据结构

考生注意：所有答案必须写在答题纸（卷）上，写在本试题上一律不给分。

一. 选择题(每题 2 分，共 30 分)

1. 算法分析的目的是 ( )。  
A. 找出数据结构的合理性  
B. 研究算法中的输入和输出关系  
C. 分析算法的效率以求改进  
D. 分析算法的易读性和文档性
2. 下列函数中渐近时间复杂度最小的是 ( )。  
A.  $T1(n) = \log_2 n + 5000n$   
B.  $T2(n) = n^2 - 8000n$   
C.  $T3(n) = n^3 + 5000n$   
D.  $T4(n) = 2n \log_2 n - 1000n$
3. 线性表的动态链表存储结构与顺序存储结构相比，优点是 ( )。  
A. 所有的操作算法实现简单  
B. 便于随机存取  
C. 便于插入与删除  
D. 便于节省存储器空间
4. 若进栈序列为 1, 2, 3, 4, 5, 6, 且进栈和出栈可以穿插进行, 则可能出现的出栈序列为 ( )。  
A. 3, 2, 6, 1, 4, 5  
B. 5, 6, 4, 2, 3, 1  
C. 5, 1, 2, 3, 4, 6  
D. 3, 4, 2, 1, 6, 5
5. 顺序存储的线性表的第一个元素的存储地址是 100, 每个元素的长度为 4, 则第 4 个元素的存储地址是 ( )。  
A. 108  
B. 112  
C. 116  
D. 120
6. 在任意一棵二叉树的先序序列和后序序列中, 各叶子之间的相对次序关系 ( )。  
A. 不一定相同  
B. 互为逆序  
C. 都不相同  
D. 都相同
7. 高度为 5 的二叉树至多有结点数为 ( )。  
A. 63  
B. 32  
C. 31  
D. 64
8. 图的邻接矩阵表示法适用于表示 ( )。  
A. 无向图  
B. 有向图  
C. 稠密图  
D. 稀疏图
9. 在一个单链表中, 若 p 所指的结点不是最后一个结点, 在 p 之后插入 s 所指的结点, 则执行 ( )。  
A.  $s \rightarrow next = p; p \rightarrow next = s$   
B.  $p \rightarrow next = s; s \rightarrow next = p$   
C.  $p = s; s \rightarrow next = p \rightarrow next$   
D.  $s \rightarrow next = p \rightarrow next; p \rightarrow next = s$
10. 若在线性表中采用折半查找法查找元素, 该线性表应该是 ( )。  
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

11. 已知一棵二叉树结点的先序序列为 ABDGCFK, 中序序列为 DGBAFCK, 则结点的后序序列为( )。  
A. GDBFKCA    B. DGBFKCA    C. KFCABDG    D. CAFKGDB
12. 对于元素是整数(占 2 个字节)的  $n$  行  $n$  列对称矩阵 A, 采用以行序为主的压缩存储方式存储到一维数组  $s[n*(n+1)/2]$  中(下三角), 若  $A[1][1]$  的起始地址是 400, 问元素  $A[8][5]$  的存储地址是( )。  
A. 432                      B. 563                      C. 484                      D. 464
13. 在所有排序方法中, 关键字的比较次数与记录的初始排列无关的是( )。  
A. Shell 排序    B. 冒泡排序    C. 直接插入排序    D. 直接选择排序
14. 具有 6 个顶点的无向图至少应有( )条边才能确保是一个连通图。  
A. 5                      B. 6                      C. 7                      D. 8
15. 如果 T2 是由树 T1 转换而来的二叉树, 那 T1 中结点的先序就是 T2 中结点的( )。  
A. 先序    B. 中序    C. 后序    D. 层次序

## 二. 填空题(每题 2 分, 共 20 分)

1. 在数据结构中, 数据的逻辑结构分\_\_\_\_\_和\_\_\_\_\_。
2. 若对关键字序列(12, 18, 4, 3, 6, 13, 2, 9, 19, 8)进行快速排序(以第一个元素为支点), 则第一趟排序得到的结果为\_\_\_\_\_。
3. 堆排序采用了\_\_\_\_\_作为其数据结构, 如果希望第一次就能找出最小关键字记录, 就建立\_\_\_\_\_堆。
4. 二叉树中度为 0 的结点数为 30, 度为 1 的结点数为 30, 总结点数为\_\_\_\_\_。
5. 向栈中压入元素的操作是先\_\_\_\_\_, 后\_\_\_\_\_。
6. 在\_\_\_\_\_的情况下, 链队列的出队操作需要修改尾指针。
7. 所谓连通图 G 的生成树, 是 G 的包含其全部  $n$  个顶点的一个极小连通子图。它必定包含且包含 G 的\_\_\_\_\_条边。
8. 对于一个有向图, 若一个顶点的度为  $k_1$ , 出度为  $k_2$ , 则对应邻接表中该顶点单链表中的边节点数为\_\_\_\_\_。
9. 设  $GetHead(p)$  为求广义表  $p$  的表头函数,  $GetTail(p)$  为求广义表  $p$  的表尾函数。其中  $()$  是函数符号, 运算  $GetTail(GetHead((a, b), (c, d, e)))$  的结果是\_\_\_\_\_。
10. 对  $n$  个结点进行快速排序, 最大比较次数是\_\_\_\_\_。

## 三. 判断题(每题 1 分, 共 10 分, 正确的选 t, 错误的选 f)

1. 一个广义表的表尾总是一个广义表。( )
2. 顺序表用一维数组作为存储结构, 因此顺序表是一维数组。( )
3. 双循环链表中, 任一结点的前驱指针均为不空。( )
4. 存储图的邻接矩阵中, 邻接矩阵的大小不但与图的顶点个数有关, 而且与图的边数也有关。( )
5. 当从一个最小堆中删除一个元素时, 需要把堆尾元素填补到堆顶位置, 然后再按条件把它逐层向下调整, 直到调整到合适位置为止。( )

6. 栈和队列都是顺序存取的线性表，但它们对存取位置的限制不同。( )
7. 一个无序的元素序列可以通过构造一棵二叉排序树而变成一个有序的元素序列。(t)
8. 一棵 m 阶 B+ 树中每个结点最多有 m 个关键码，最少有 2 个关键码。( )
9. 拓扑排序是一种内部排序的算法。( )
10. 空串与空格相同。( )

#### 四. 简答题 (50 分)

1. 对关键字序列 (49, 38, 65, 97, 75, 13, 27, 51, 55, 10) 进行一趟希尔排序(由小到大)。试写出第 1 趟 (增量  $d_1=5$ ) 希尔排序的结果及元素移动次数。(4 分)
2. 已知一个图如图 1 所示, (10 分)
  - (1) 请写出其邻接矩阵和邻接表。
  - (2) 请写出其拓扑排序序列。

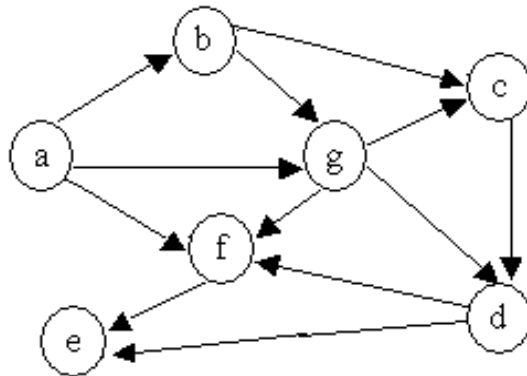


图 1

3. 在图 2 所示的 AOE 网中, 试找出此网络中的关键活动和关键路径。(10 分)

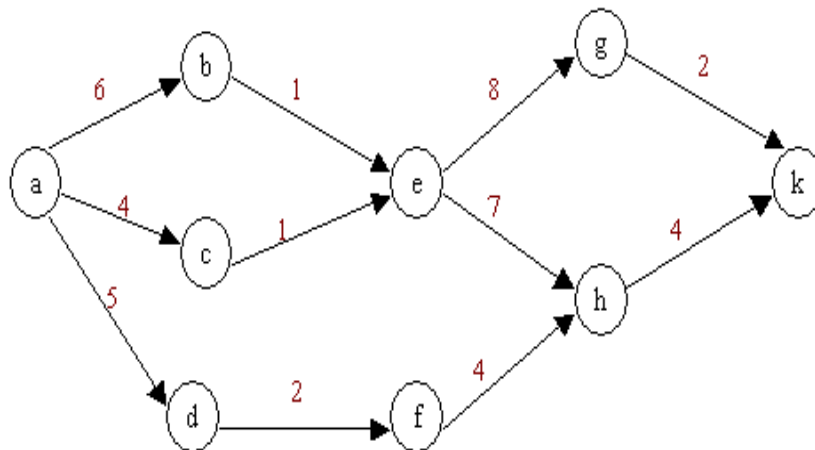


图 2

4. 已知一颗3阶的B-树如图3所示,若删除44和79之后,画出这棵B-树的最终状态。(8分)

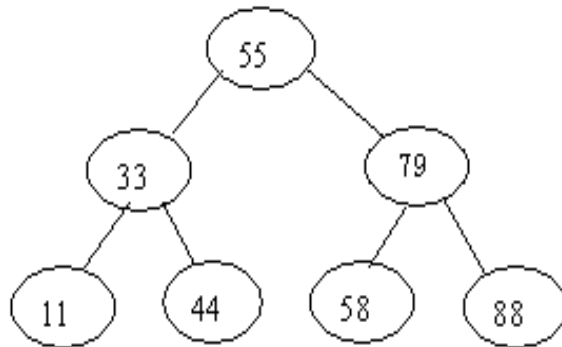


图3

5. 设有一段正文是由字符集{A, B, C, D, E, F}组成的,正文长度为100个字符,其中每个字符在正文中出现的次数分别为17, 12, 5, 28, 35, 3。若采用 Huffman 编码对这段正文进行压缩存储,请完成如下工作:(10分)
- (1) 构造出 Huffman 树(规定权值较小的结点为左子树);
  - (2) 写出每个字符的 Huffman 编码;
  - (3) 计算按 Huffman 编码压缩存储这段正文共需要多少个字节(设每个字节为8位二进制位组成);
  - (4) 若有另一段正文的二进制编码序列为01101010110011,请用(2)的 Huffman 编码将它翻译成所对应的正文。
6. 设有一组关键字(47, 7, 29, 11, 16, 92, 22, 8, 3)采用散列函数  $H(\text{key}) = \text{key} \% 11$ , 开放地址的线性探测再散列方法解决冲突,试在0~10的散列地址空间中对该关键字序列(按从左到右的次序)构造散列表,并计算在查找概率相等的前提下,成功查找的平均查找长度。(8分)

### 五. 算法填空,(每空2分,共16分)

1. 下面的算法是一个在元素按值递增排列,并以带头结点的单链表作存储结构的线性表中,删除表中所有值大于  $\text{mink}$  且小于  $\text{maxk}$  的元素(若表中存在这样的元素),同时释放被删除结点空间。请在\_\_\_\_\_处填上适当内容,使其成为一个完成算法。

```

Void delmn(LinkList &L, int mink, int maxk)
{
  LinkList p=L,q,s;
  if ((p->next)&&(mink<=maxk))
  { while (_____ (1) _____) p=p->next; }
  if(_____ (2) _____)
  { q=p->next;
    while(q->data<maxk)
    { s=q; q=q->next; free(s); }
    _____ (3) _____
  }
}

```

2. 下面是一个图的广度优先非递归算法，请在\_\_\_\_\_处填上适当内容，使其成为一个完成算法。

```
void BFSTraverse (Graph G, Status(*Visit) (VertexTyp e))
{ for(v=0; v<G.vexnum; v++)  visited[v]=FALSE;
  InitQueue(Q);
  for(v=0; _____ (4) _____ v++)
  { if(!visited[v])
    { visited[v]=TRUE;
      _____ (5) _____
      EnQueue(&Q,v);
      while( _____ (6) _____ )
      { _____ (7) _____
        for(w=FirstAdjVex(G,u); w>=0; w=NextAdjVex(G,u,w))
        { if(!visited[w])
          { visited[w]=TRUE;
            visit(w);
            _____ (8) _____
          }
        }
      }
    }
  }
}
```

## 六. 编写算法 (24)

1. 试编写出后序遍历二叉树的算法 (10分)
2. 已知  $n$  个顶点的带权图用邻接矩阵表示，试编写算法实现用 Kruskal 算法构造最小生成树。(14分)