

杭州电子科技大学

2011 年攻读硕士学位研究生入学考试

《数据结构》试题

(试题共 五 大题, 5 页, 150 分)

姓名_____报考专业_____准考证号_____

【所有答案必须写在答题纸上, 做在试卷或草稿纸上无效!】

一、选择题(每小空 2 分, 共 28 分)

1. 在下列数据结构中_____具有先进先出特性, _____具有先进后出特性。
a: 线性表 b: 栈 c: 队列 d: 串
2. 如下关于串的陈述中, 正确的是_____、_____。
a: 串是数据元素类型特殊的线性表 b: 串中的元素是字母
c: 串中若干个元素构成的子序列称为子串 d: 空串即为空格串
3. 对广义表 $A = (((a), (b)), ((c)))$
执行操作 $\text{gettail}(\text{gethead}(\text{gettail}(A)))$ 的结果是: _____。
执行操作 $\text{gethead}(\text{gettail}(\text{gethead}(A)))$ 的结果是: _____。
a: $()$ b: $(())$ c: (a) d: (b) e: (c)
4. 任何一个连通网的最小生成树_____。
a: 只有一棵 b: 有一棵或多棵 c: 一定有多棵 d: 可能不存在
5. 在有 n 个结点的二叉树的三叉链表表示中, 空指针数为 _____。
a: 不确定 b: n c: $n+1$ d: $n+2$
6. 关键路径是指在只有一个源点和一个汇点的有向无环网中源点至汇点_____的路径。
a: 弧的数目最多 b: 弧的数目最少 c: 权值之和最大 d: 权值之和最小
7. 设无向图 $G = (V, E)$ 和 $G' = (V', E')$, 若 G' 是 G 的生成树,
则下面不正确的说法是_____。
a: G' 是 G 的子图 b: G' 是 G 的连通分量
c: G' 是 G 的无环子图 d: G' 是 G 的极小连通子图且 $V' = V$
8. 下列查找方法中_____适用于查找单链表。
a: 顺序查找 b: 折半查找 c: 分块查找 d: hash 查找
9. 下列排序方法中, _____是稳定的; _____具有最好的平均性能; 当待排序序列的关键字次序为倒序时, 若需为之进行正序排序, 下列方案中以_____为佳。
a: 堆排序 b: 快速排序 c: 直接插入排序 d: 简单选择排序

二、填空题（每空 2 分，共 26 分）

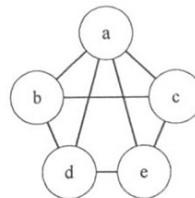
1. 数据结构通常有下列 4 类基本结构：线性结构、树型结构、图型结构、_____。
2. 线性表的_____存储结构是以物理位置来表示数据元素之间的逻辑关系的。
而线性表的_____存储结构是通过指针保持数据元素之间的逻辑关系的。
3. n 个顶点的强连通图至少有_____条弧，至多有_____条弧。
4. 若某一二叉树按中序遍历可得到有序序列，则该二叉树是_____。若某一二叉树从根结点到其它任一结点的路径上所经过的结点序列按其关键字递增有序，则该二叉树是_____。
5. 若对完全二叉树中的结点从 1 开始按层进行编号，设最大编号为 n ，则编号为 i 的结点 ($1 < i \leq n$) 的父结点编号为_____；所有编号_____的结点为叶子结点。
6. 压栈次序为 a 、 b 、 c ，则不可能得到的输出序列是_____。
7. 已知待排序序列为：33, 34, 7, 28, 38, 11, 65, 15, 37, 20。则：
以第一个元素为枢轴的快速排序一趟分划的结果是_____。
堆排序初始建堆(小顶堆)的结果是_____。
希尔排序第一趟(增量为 3)的结果是_____。

三、图示结构题（每小题 8 分，共 40 分）

1. 已知某二叉树的先序遍历次序为：ABCDEFGH，中序遍历次序为：BADCFEGH。
(1) 画出该二叉树形。
(2) 给出该二叉树的后序遍历次序。
(3) 画出中序线索化后的二叉树形。

2. 已知某无向图如右图所示：

- (1) 画出该图的邻接表存储结构。
- (2) 画出该图的邻接矩阵存储结构。
- (3) 根据你所绘制的邻接表给出 DFS 及 BFS 次序。



3. 依序将关键字 20、40、30、80、70、50、60、10 插入到一棵 2-3 树中(初始状态为空)，
(1) 请画出该 B-树。
(2) 再先后删除关键字 40、60，画出删除后的 B-树。
4. 设哈希函数为 $H(\text{key}) = \text{key} \bmod 7$ ，用链地址法处理冲突，若依次在哈希表中插入 12 个元素 32、65、83、25、74、21、33、18、61、27、47、28。
(1) 画出它们在表中的分布情形。
(2) 计算其平均成功的查找长度。
5. 假设用于通讯的电文仅由 8 个字符 A、B、C、D、E、F、G、H 组成，字符在电文中出现的频率分别为 3、12、9、23、2、17、21、13
(1) 画出你所建的哈夫曼树，
(2) 给出每一字符的哈夫曼编码。

四、阅读以下函数，指出算法的功能（每小题 6 分，共 36 分）

1. Status A1(SqList L, ElemType cur_e, ElemType &next_e)
{ // 初始条件：顺序线性表 L 已存在
int i=1;
ElemType *p=L.elem;
while(i < L.length && *p != cur_e){
i++;
p++;
}
if(i == L.length)
return INFEASIBLE;
else{
next_e = *++p;
return OK;
}
}
2. Status A2(LinkList L, int i, ElemType e)
{// 初始条件：带头结点的单链表 L 已存在
int j = 0;
LinkList p = L, s;
while(p && j < i-1){
p = p->next;
j++;
}
if(!p || j > i-1)
return ERROR;
s = (LinkList)malloc(sizeof(LNode));
s->data = e;
s->next = p->next;
p->next = s;
return OK;
}
3. int A3(LinkQueue Q)
{ // 初始条件：链队列 Q 已存在
int i = 0;
QueuePtr p;
P = Q.front;

```

        while(Q.rear != p) {
            i++;
            p = p->next;
        }
        return i;
    }

4. void A4(BiTree T, Status(*Visit)(TElemType))
    { // 初始条件: 二叉树 T 已存在
      if(T) {
          A4(T->lchild, Visit);
          A4(T->rchild, Visit);
          Visit(T->data);
      }
    }

5. int A5(SSTable ST, KeyType key)
    { //初始条件:顺序表 ST 已存在
      int i;
      ST.elem[0].key=key;
      for(i = ST.length; ! EQ(ST.elem[i].key, key); --i);
      return i;
    }

6. int A6(SqList L, int i)
    { //初始条件:顺序表 L 已存在
      int min;
      int j, k;
      k = i;
      min = L.r[i].key;
      for(j = i + 1; j <= L.length; j++)
          if(L.r[j].key < min) {
              k = j;
              min = L.r[j].key;
          }
      return k;
    }

```

五、算法设计题（每小题 10 分，共 20 分）

1. 设单链表结点结构为：

```
typedef struct LNode {  
    int      data;  
    struct LNode *next;  
}*LinkList;
```

写一函数 void A7 (LinkList L)

试采用直接插入排序的方法将单链表 L （带头结点）中的结点按非递减次序排列。

2. 设二叉链表结构为：

```
typedef struct BiTNode {  
    TElemType      data;  
    struct BiTNode *lchild, *rchild;  
} *BiTree;
```

写一函数 void A8 (BiTree T) 求二叉树 T 的深度。