

华中师范大学 2007-2008 学年第 1 学期

期末考试试卷 (A 卷)

课程名称 数据结构 课程编号 84810007 任课教师 魏开平、王敬华、沈显君、王明安、彭熙

题型	单选题	填空题	简答题	阅读题	编程题	总分
分值	20	16	24	20	20	100
得分						

得分	评阅人

一、选择题 (每小题 1 分, 共 20 题)

- 某程序的时间复杂度为 $(3n + n \log_2 n + n^2 + 8)$, 其数量级表示为 ()。
A. $O(n)$ B. $O(n \log_2 n)$ C. $O(n^2)$ D. $O(\log_2 n)$
- 队列的插入操作是在 () 进行。
A. 队首 B. 队尾 C. 队前 D. 对后
- 二叉树上叶结点数等于 ()。
A. 分支结点数加 1 B. 单分支结点数加 1 C. 双分支结点数加 1 D. 双分支结点数减 1
- 每次从无序表中取出一个元素, 把它插入到有序表中的适当位置, 此种排序方法叫做 () 排序。
A. 插入 B. 交换 C. 选择 D. 归并
- 在一个图中, 所有顶点的度数之和等于所有边数的 () 倍。
A. 2 B. 1 C. 3 D. 4
- 队列的删除操作是在 () 进行。
A. 队首 B. 队尾 C. 队前 D. 对后
- 当利用大小为 N 的数组顺序存储一个栈时, 假定用 $top=0$ 表示栈空, 则退栈时, 用 () 语句修改 top 指针。
A. $top++;$ B. $top=0;$ C. $top--;$ D. $top=N;$
- 由权值分别为 3, 6, 7, 2, 5 的叶子结点生成一棵哈夫曼树, 它的带权路径长度为 ()。
A. 51 B. 23 C. 53 D. 74
- 在一棵二叉树中, 第 4 层上的结点数最多为 ()。
A. 31 B. 8 C. 15 D. 16
- 向堆中插入一个元素的时间复杂度为 ()。
A. $O(\log_2 n)$ B. $O(n)$ C. $O(1)$ D. $16 O(n \log_2 n)$
- 在一个长度为 n 的顺序存储的线性表中, 向第 i 个元素 ($1 \leq i \leq n+1$) 之前插入一个新元素时, 需要从后向前依次后移 () 个元素。
A. $n-i$ B. $n-i+1$ C. $n-i-1$ D. i

12. 在线性表的散列存储中, 若用 m 表示散列表的长度, n 表示待散列存储的元素的个数, 则 α 等于 ()。
- A. n/m B. m/n C. $n/(n+m)$ D. $m/(n+m)$
13. 在图的邻接表表示中, 每个结点所指向链表中的边结点都具有相同的 ()。
- A. 进入结点号 B. 离开结点号 C. 边号 D. 其他意义
14. 在一个具有 n 个顶点的无向图中, 要连通所有顶点则至少需要 () 条边。
- A. n B. $2n$ C. $n-1$ D. $n+1$
16. 若需要利用形参直接访问实参, 则应把形参变量说明为 () 参数。
- A. 指针 B. 引用 C. 值 D. 常量
17. 在一个单链表 HL 中, 若要在指针 q 所指结点的后面插入一个由指针 p 所指向的结点, 则执行
- A. $q \rightarrow \text{next} = p \rightarrow \text{next}; p \rightarrow \text{next} = q;$ B. $p \rightarrow \text{next} = q \rightarrow \text{next}; q \rightarrow \text{next} = p;$
- C. $q \rightarrow \text{next} = p \rightarrow \text{next}; p \rightarrow \text{next} = q;$ D. $p \rightarrow \text{next} = q \rightarrow \text{next}; q \rightarrow \text{next} = p;$
18. 在一个顺序队列中, 队首指针指向队首元素的 () 位置。
- A. 后一个 B. 前一个 C. 当前 D. 与它无关
19. 向二叉搜索树中插入一个元素时, 其时间复杂度大致为 ()。
- A. $O(\log n)$ B. $O(n)$ C. $O(1)$ D. $O(n \log n)$
20. 下面程序段的时间复杂度为 _____。
- ```
for(int i=0; i<m; i++)
 for(int j=0; j<n; j++)
 a[i][j]=i*j;
```
- A.  $O(m^2)$       B.  $O(n^2)$       C.  $O(m*n)$       D.  $O(m+n)$

| 得分 | 评阅人 |
|----|-----|
|    |     |

### 二、填空题 (每空 2 分, 共 16 分)

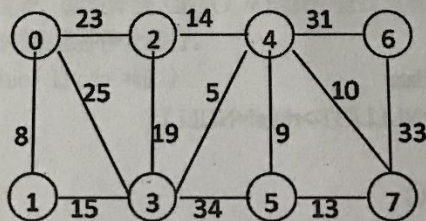
- 假定一棵树的广义表表示为  $A(B(C(D,E), F, G(H,I,J)), K)$ , 则度为 2 的结点数分别为 \_\_\_\_\_。
- 在一个具有  $n$  个顶点的有向完全图中, 包含有 \_\_\_\_\_ 条边。
- 从一棵二叉搜索树中查找一个元素时, 若元素的大于根结点的值, 则继续向 \_\_\_\_\_。
- 对于一个顺序存储的线性表, 在表尾插入元素的时间复杂度为 \_\_\_\_\_。
- 后缀算术表达式  $248+3*497-*/+$  所对应的中缀算术表达式为 \_\_\_\_\_。
- 在一棵二叉搜索树中, 每个分支结点的左子树上所有的结点的值一定 \_\_\_\_\_ 该结点的值。
- 假定一组记录的排序码为  $(46, 79, 56, 38, 40, 80)$ , 取第一个为基准, 对其进行快速排序的第一次扫描后, 序列为 \_\_\_\_\_。
- 对于线性表  $(18, 25, 63, 50, 41, 32, 90, 66)$  进行散列存储时, 若选用  $H(K) = K \% 11$  作为散列函数, 则散列表中地址为 3 的元素有 \_\_\_\_\_ 个。



| 得分 | 评阅人 |
|----|-----|
|    |     |

### 三. 简答题。(每小题 6 分, 共 24 分)

1. 空堆开始依次向堆中插入线性表  $(64, 52, 12, 48, 45, 26)$  中的每个元素, 请以线性表的形式给出每插入一个元素后堆的状态。(为小根堆)
2. 在一份电文中共使用五种字符: A, G, F, U, Y, Z, 它们的出现频率依次为 12, 9, 18, 7, 14, 11, 求出每个字符的哈夫曼编码。
3. 假定一个待散列存储的线性表为  $(32, 75, 29, 63, 48, 94, 25, 46, 18, 70)$ , 散列地址空间为  $HT[13]$ , 若采用除留余数法构造散列函数和线性探查法处理冲突, 试求出每一元素的散列地址, 画出最后得到的散列表, 求出平均查找长度。
4. 对于下图, 若按照克鲁斯卡尔算法产生最小生成树, 写出得到的各条边的次序。



| 得分 | 评阅人 |
|----|-----|
|    |     |

四. 阅读程序写结果 (每小题 5 分, 共 20 分)

```
1. int AA(LNode *HL, int x)
{
 int n=0;
 LNode *p=HL;
 while (p!=NULL)
 {
 if (p->data==x) n++;
 p=p->next;
 }
 return n;
}
```

对于结点类型为 LNode 的单链表, 该函数的功能是: \_\_\_\_\_。

```
2. void BB (LNode *HL, int item)
{
 LNode *newptr=new LNode;
 newptr->data=item;
 LNode *p=HL;
 while (p->next!=HL) p=p->next;
 newptr->next=HL;
 p->next=newptr;
}
```

对于结点类型为 LNode 的单链表, 该函数的功能是: \_\_\_\_\_。

```
3. void CC (BTreeNode *BT)
{
 if(BT!=NULL)
 {
 cout<<BT->data;
 if(BT->left!=NULL||BT->right!=NULL)
 {
 cout<<'(';
 CC(BT->left);
 if(BT->right!=NULL) cout<<',';
 CC(BT->right);
 cout<<')';
 }
 }
}
```

该函数的功能是: \_\_\_\_\_。

4. void DD(int A[],int n)

```
{
 int x;
 int i,j,flag;
 for(i=1;i<n-1;i++){
 flag=0;
 for(j=n-1;j>=i;j--){
 if(A[j]<A[j-1]){ x=A[j]; A[j]=A[j-1]; A[j-1]=x; flag=1; }
 }
 if(flag==0) return;
 }
}
```

执行该函数调用后得到的输出结果是：\_\_\_\_\_。

| 得分 | 评阅人 |
|----|-----|
|    |     |

五. 编写程序（函数）（每小题 10 份，共 20 分）

1. 编写一个算法，求二叉树中的最小的关键字的值，关键字的类型为整形。  
函数原形：int MinValue ( BTree T )

2. 对于结点类型为 LNode 的单链表，编写算法（函数）从链表中查找出所有元素的最大值，该值由函数返回，若单链表为空，则显示出错信息并停止运行。  
函数原形：ElemType MaxValue( LNode \*HL )