

试卷代号:1010

座位号

中央广播电视大学 2010—2011 学年度第一学期“开放本科”期末考试

数据结构 试题

2011 年 1 月

题 号	一	二	三	四	五	总 分
分 数						

得 分	评卷人

一、单项选择题,在括号内填写所选择的标号(每小题 2 分,共 18 分)

1. 执行下面程序段时,S 语句的执行次数为()。

```
for(int i=1; i<=n; i++)  
    for(int j=1; j<=i; j++) S;
```

- A. n^2
- B. $n^2/2$
- C. $n(n+1)$
- D. $n(n+1)/2$

2. 二维数组实际上是由嵌套的()实现的。

- A. 一维数组
- B. 多项式
- C. 三元组表
- D. 简单变量

3. 对于表头指针为 first 的单链表,其空表的判定条件是()。

- A. `first == NULL;`
- B. `first->link == NULL;`
- C. `first->link == first;`
- D. `first! = NULL;`

4. 若让元素 1,2,3 依次进栈,则出栈次序不可能出现()种情况。
- A. 3,2,1
 - B. 2,1,3
 - C. 3,1,2
 - D. 1,3,2
5. 在一棵具有 n 个结点的满二叉树中,共包含有()个分支结点。
- A. $n-1$
 - B. $n/2$
 - C. $n/2+1$
 - D. $n/2-1$
6. 若搜索每个元素的概率相等,则在长度为 n 的顺序表上搜索任一元素的平均搜索长度为()。
- A. n
 - B. $n+1$
 - C. $(n-1)/2$
 - D. $(n+1)/2$
7. 向一棵二叉搜索树插入一个元素后,该树中的叶子结点数比插入前一定()。
- A. 增加
 - B. 减少
 - C. 相等
 - D. 不减少
8. 为了实现图的广度优先搜索遍历,其算法使用的一个辅助数据结构是()。
- A. 栈
 - B. 队列
 - C. 二叉树
 - D. 树
9. 在一棵 5 阶 B 树中,每个结点最多允许有()个关键码。
- A. 2
 - B. 3
 - C. 4
 - D. 5

得 分	评卷人

二、判断题,在每小题后面的括号内打对号(√)表示叙述正确或打叉号(×)表示叙述错误(每小题 2 分,共 14 分)

10. 若每次从队列中取出的是具有最高优先权的元素,则称此队列为优先级队列。 ()
11. 递归定义的数据结构通常不需要采用递归的算法对其运算。()
12. 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。()
13. 对于一棵具有 n 个结点、高度为 h 的二叉树,进行任一种次序遍历的时间复杂度均为 $O(n)$ 。()
14. 对于同一组数据元素,生成二叉搜索树的形态与插入元素的次序无关。()
15. 装载因子是散列存储中的一个重要指标,它不能反映散列表的装满程度。()
16. 在一棵 B 树中,所有叶结点都处在同一层上。()

得 分	评卷人

三、填空题,在横线处填写合适的内容(每小题 2 分,共 14 分)

17. 在类的继承结构中,位于上层的类叫做基类或父类,而位于下层的类叫做派生类或_____类。
18. 设链栈中结点的结构为(data,link),栈顶指针为 top,当从该链栈删除一个结点时,应把_____的值赋给 top。
19. 广义表的_____定义为广义表中括号被嵌套的最大重数。
20. 在一棵高度为 3 的完全二叉树中,最少含有_____个结点。假定树根结点的高度为 0。
21. 从有序表(12,18,30,43,56,78,82,95)中折半搜索 56 元素时,其搜索长度为_____。
22. 具有 n 个顶点的连通图中至少包含有_____条边。
23. 假定一个数据集合为{46,79,56,38,40,84},则在构成的最大堆(即大根堆)中,其堆顶元素为_____。

得分	评卷人

四、运算题(每小题 8 分,共 40 分)

24. 假定一棵二叉树的广义表表示为 $A(B(D(G)),C(E,F))$, 分别写出对它进行先序、中序和按层遍历的结果。

先序:

中序:

按层:

25. 已知一个有序表 $(15, 26, 34, 39, 45, 56, 58, 63, 74, 76, 83, 94)$ 顺序存储于一维数组 $a[12]$ 中, 根据折半搜索过程填写成功搜索下表中所给元素 34、56、74、83 时的比较次数。

元素	34	56	74	83
比较次数				

26. 假定一个线性表为 $(56, 27, 34, 95, 73, 16, 50, 62)$, 根据此线性表中元素次序生成一棵二叉搜索树, 分别求出该二叉搜索树中的单分支结点数和双分支结点数。

单分支结点数:

双分支结点数:

27. 已知一个带权图的顶点集 V 和边集 G 分别为:

$$V = \{0, 1, 2, 3, 4, 5\};$$

$$E = \{(0, 1)19, (0, 2)21, (0, 3)14, (1, 2)16, (1, 5)5, (2, 4)11, (3, 4)18, (4, 5)6\};$$

试根据普里姆算法, 从顶点 1 出发, 求出其最小生成树, 在下面横线上填写依次得到的最小生成树中的每条边。

_____ , _____ , _____ , _____ , _____ 。

28. 设散列表的长度 $m=7$; 散列函数为 $H(K)=K \bmod m$, 给定的关键码序列为 $\{19, 14, 23, 40, 69\}$, 并假定采用的闭散列表为 $HT[m]$, 采用的解决冲突的方法为线性探查法, 求出在最后得到的散列表中, 关键码 19、14 和 69 的存储位置和对应的查找长度。

元素:	19	14	69
存储位置:			
查找长度:			

得 分	评卷人

五、算法分析题(每小题 7 分,共 14 分)

29. 设 rear 是以循环链表表示的队列的队尾指针, EnLQueue 函数实现把 x 插入到队尾的操作。阅读下面算法, 在划有横线的上面填写合适的内容。

```
void EnLQueue(ListNode * & rear, ElemType x)
{
    ListNode * p;
    p = new ListNode;      //p 指向动态分配的结点空间
    p->data = x;
    p->link = _____;
    rear->link = p;
    _____;
};
```

30. 已知二叉树中的结点类型 BinTreeNode 定义为:

```
struct BinTreeNode {ElemType data; BinTreeNode * left, * right;};
```

其中 data 为结点值域, left 和 right 分别为指向左、右子女结点的指针域, 根据下面算法定义写出算法的功能。在此算法中, 参数 BT 为初始指向一棵二叉树的树根指针。

```
int BTreeLeafCount(BinTreeNode * BT)
{
    if(BT == NULL) return 0;
    else if(BT->left == NULL && BT->right == NULL) return 1;
    else return BTreeLeafCount(BT->left) + BTreeLeafCount(BT->right);
}
```

算法功能:

试卷代号:1010

中央广播电视大学 2010—2011 学年度第一学期“开放本科”期末考试

数据结构 试题答案及评分标准

(供参考)

2011 年 1 月

一、单项选择题,在括号内填写所选择的标号(每小题 2 分,共 18 分)

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

二、判断题,在每小题后面的括号内打对号(√)表示叙述正确或打叉号(×)表示叙述错误(每小题 2 分,共 14 分)

10. √(对) 11. ×(错) 12. √(对) 13. √(对) 14. ×(错)
15. ×(错) 16. √(对)

三、填空题,在横线处填写合适的内容(每小题 2 分,共 14 分)

17. 子(继承)
18. top- \rightarrow link
19. 深度
20. 8
21. 3
22. n-1
23. 84

四、运算题(每小题 8 分,共 40 分)

24. 先序:A,B,D,G,C,E,F //3 分
 中序:B,G,D,A,E,C,F //3 分
 按层:A,B,C,D,E,F,G //2 分
25. 元素 34 56 74 83
 比较次数 2 1 2 3
 //得分 2 分 2 分 2 分 2 分

26. 单分支结点数:3 //4分

双分支结点数:2 //4分

27. (1,5)5, (5,4)6, (4,2)11, (4,3)18, (3,0)14

//得分:2分 2分 2分 1分 1分

28.

元素:	19	14	69
存储位置:	5	0	1
查找长度:	1	1	3
得分:	3分	3分	2分

五、算法分析题(每小题7分,共14分)

29. rear->link、rear=p //第1个空4分,第2个空3分

30. 求出并返回树根指针为BT的一棵二叉树中叶子结点的总数。 //7分