

## 哈尔滨工业大学

## 2013 年春季学期数据结构与算法 试卷 A

班级	
学号	
姓名	

题号	一	二	三		总分
分数	24	46	30		100
得分					

## 一、选择题（每小题 2 分，共 24 分）

1. 下列程序段的时间复杂度为（ ）。

```
i = 0; s = 0;
while (s < n){
    s = s + i;
    i++;
}
```

- A.  $O(n^{1/2})$       B.  $O(n^{1/3})$       C.  $O(n)$       D.  $O(n^2)$

2. 设某链表中最常用的操作是在链表的尾部插入或删除元素，则选用下列（ ）存储方式最节省运算时间。

- A. 单向链表      B. 单向循环链表      C. 双向链表      D. 双向循环链表

3. 设指针 q 指向单项链表中结点 A，指针 p 指向单链表中结点 A 的后继结点 B，指针 s 指向被插入的结点 X，则在结点 A 和结点 B 之间插入结点 X 的操作序列为（ ）。

- A.  $s \rightarrow next = p \rightarrow next$ ;  $p \rightarrow next = s$ ;      B.  $q \rightarrow next = s$ ;  $s \rightarrow next = p$ ;  
C.  $p \rightarrow next = s \rightarrow next$ ;  $s \rightarrow next = p$ ;      D.  $p \rightarrow next = s$ ;  $s \rightarrow next = q$ ;

4. 设输入序列为 1、2、3、4、5、6，则通过栈的作用后可以得到的输出序列为（ ）。

- A. 5、3、4、6、1、2      B. 3、2、5、6、4、1  
C. 3、1、2、5、4、6      D. 1、5、4、6、2、3

5. 设有一个 10 阶的下三角矩阵 A(包括对角线)，按照从上到下、从左到右的顺序存储到连续的 55 个存储单元中，每个数组元素占一个字节存储空间，则  $A[5][4]$  地址与  $A[0][0]$  的地址之差为（ ）。

- A. 10      B. 19      C. 28      D. 55

6. 设一棵 m 叉树中有  $N_1$  个度为 1 的结点， $N_2$  个度为 2 的结点，…， $N_m$  个度为 m 的结点，则该树中共有（ ）个叶子结点。

- A.  $\sum_{i=1}^m (i-1)N_i$       B.  $\sum_{i=1}^m N_i$       C.  $\sum_{i=2}^m N_i$       D.  $1 + \sum_{i=2}^m (i-1)N_i$

7. 二叉排序树中左子树上所有结点的值均为（ ）根结点的值。

- A. <      B. >      C. =      D. !=

8. 设一组权值集合  $W = \{15, 3, 14, 2, 6, 9, 16, 17\}$ ，要求根据这些权值集合构建一棵哈弗曼树，则这棵哈弗曼树的带权路径长度为（ ）。

- A. 129      B. 219      C. 189      D. 229



9. 设有  $n$  个关键字具有相同的 Hash 函数值，则用线性探测法把这  $n$  个关键字映射到 Hash 表中需要做（ ）次线性探测。

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

10. 设某棵二叉树中只有度为 0 和度为 2 的结点且度为 0 的结点数为  $n$ ，则这棵二叉树中共有（ ）个结点。

- A.  $2n$                       B.  $n+1$                       C.  $2n-1$                       D.  $2n+1$

11. 设一组初始记录关键字的长度为 8，则最多经过（ ）趟插入排序可以得到有序序列。

- A. 6                      B. 7                      C. 8                      D. 9

12. 设一组初始记录关键字序列为(Q,H,C,Y,P,A,M,S,R,D,F,X)，则按字母升序的第一趟冒泡排序结束后的结果是（ ）。

- A. F, H, C, D, P, A, M, Q, R, S, Y, X                      B. P, A, C, S, Q, D, F, X, R, H, M, Y  
C. A, D, C, R, F, Q, M, S, Y, P, H, X                      D. H, C, Q, P, A, M, S, R, D, F, X, Y

## 二、填空题（每空 2 分，共 46 分）

1. 设需要对 5 个不同的记录关键字进行排序，则至少需要比较\_\_\_\_\_次，至多需要比较\_\_\_\_\_次。

2. 快速排序算法的平均时间复杂度为\_\_\_\_\_，直接插入排序算法的平均时间复杂度为\_\_\_\_\_。

3. 设二叉排序树的高度为  $n$ ，则在该树中查找关键字  $key$  最多需要比较\_\_\_\_\_次。

4. 设在长度为 20 的有序表中进行二分查找，则比较一次查找成功的结点数有\_\_\_\_\_个，比较两次查找成功的结点数有\_\_\_\_\_个。

5. 设一棵  $m$  叉树的结点数为  $n$ ，用多重链表表示其存储结构，则该树中有\_\_\_\_\_个空指针域。

6. 设指针变量  $p$  指向单链表中结点 A，则删除 A 的语句序列为： $q = p \rightarrow next$ ； $p \rightarrow data = q \rightarrow data$ ； $p \rightarrow next =$ \_\_\_\_\_； $free(q)$ ；。

7. 数据结构从逻辑上划分为三种基本类型：\_\_\_\_\_、\_\_\_\_\_、\_\_\_\_\_。

8. 设无向图  $G$  中有  $n$  个顶点  $e$  条边，则用邻接矩阵作为图的存储结构进行深度优先或广度优先遍历时的时间复杂度为\_\_\_\_\_；用邻接表作为图的存储结构进行深度优先或广度优先遍历的时间复杂度为\_\_\_\_\_。

9. 设散列表的长度为 8，散列函数  $H(k)=k\%7$ ，用线性探测法解决冲突，则根据一组初始关键字序列(8,15,16,22,30,32)构造出的散列表的平均查找长度是\_\_\_\_\_。

10. 设一组初始关键字序列为(38,65,97,76,13,27,10)，则第 3 趟冒泡排序结束后的结果为\_\_\_\_\_。

11. 设一组初始关键字序列为(38,65,97,76,13,27,10)，则第 3 趟简单选择排序结束后的结果为\_\_\_\_\_。



12. 设有向图 G 中的有向边的集合  $E=\{<1,2>, <2,3>, <1,4>, <4,5>, <5,3>, <4,6>, <6,5>\}$ , 则该图的一个拓扑序列为\_\_\_\_\_。

13. 下面程序段的功能是建立二叉树的算法，请在下划线处填上正确的内容。

```
typedef struct node{
    int data;
    struct node* lchild;
    _____;
}BiTree;
void CreateBiTree(BiTree *&bt){
    scanf("%c", &ch);
    if (ch == '#')
        _____;
    else{
        bt = (BiTree*)malloc(sizeof(BiTree));
        bt->data = ch;
        _____;
        CreateBiTree(bt->rchild);
    }
}
```

14. 下面程序段的功能是利用从尾部插入的方法建立单链表的算法，请在下划线处填上正确的内容。

```
typedef struct node{
    int data;
    struct node *next;
}lklist;
void lklistcreate(_____*&head){
    for (i = 1; i <= n; i++){
        p = (lklist*)malloc(sizeof(lklist));
        scanf("%d", &(p->data));
        p->next = NULL;
        if (i == 1)
            head = q = p;
        else{
            q->next = p;
            _____;
        }
    }
}
```

### 三、算法设计题（共 30 分）

1. 设计在链式存储结构上合并排序的算法。
2. 设计在二叉排序树上查找结点 X 的算法。
3. 设关键字序列  $(k_1, k_2, \dots, k_{n-1})$  是堆，设计算法将关键字序列  $(k_1, k_2, \dots, k_{n-1}, X)$  调整为堆。