

# 启航计算机 VIP 班习题课数据结构测试

## 一、选择题

1. 下面程序的时间复杂度为( )。

```
For(int i=1,s=0;i<=n;i++){  
    int t=1;  
    for(int j=1;j<=i;j++){  
        t=t*j;  
    }  
    s=s+t;  
}
```

A.  $O(n)$ ;                      B.  $O(n^2)$ ;                      C.  $O(n^3)$ ;                      D.  $O(n^4)$ 。

2. 若某链表最常用的操作是在末尾插入结点和删除尾结点, 则该链表最节省时间的存储方式是( )。

A. 单链表;    B. 单循环链表;    C. 带尾结点的单循环链表;    D. 带头结点的双循环链表。

3. 若线性表中总的元素个数基本稳定, 但经常要在表头删除元素, 在表尾插入元素, 那么最好采用( )实现该线性表。

A. 带头指针的单链表;    B. 双向循环链表;    C. 循环顺序队列;    D. 顺序表。

4. 在双向链表指针  $p$  的结点前插入一个指针  $q$  的结点操作是( )。

A.  $p \rightarrow \text{prior} = q; q \rightarrow \text{next} = p; p \rightarrow \text{prior} \rightarrow \text{next} = q; q \rightarrow \text{prior} = q;$

B.  $p \rightarrow \text{prior} = q; p \rightarrow \text{prior} \rightarrow \text{next} = q; q \rightarrow \text{next} = p; q \rightarrow \text{prior} = p \rightarrow \text{prior};$

C.  $q \rightarrow \text{next} = p; q \rightarrow \text{prior} = p \rightarrow \text{prior}; p \rightarrow \text{prior} \rightarrow \text{next} = q; p \rightarrow \text{prior} = q;$

D.  $q \rightarrow \text{prior} = p \rightarrow \text{prior}; q \rightarrow \text{next} = q; p \rightarrow \text{prior} = q; p \rightarrow \text{prior} = q;$

5. 若长度为  $n$  的线性表采用顺序存储结构, 在表的第  $i$  个位置插入一个数据元素, 需要移动表中( )个数据元素;

A.  $i$ ;                      B.  $n+i$ ;                      C.  $n-i+1$ ;                      D.  $n-i-1$ 。

6. 采用顺序存储结构的线性表的插入算法中, 当  $n$  个空间已满时, 可申请再增加分配  $m$  个空间。若申请失败, 则说明系统没有( )可分配的存储空间。

A.  $m$  个;                      B.  $m$  个连续的;                      C.  $n+m$  个;                      D.  $n+m$  个连续的。

7. 若用一个大小为 6 的数组来实现循环队列, 且当前  $\text{rear}$  和  $\text{front}$  的值分别为 0 和 3, 当从队列中删除一个元素, 再加入两个元素后,  $\text{rear}$  和  $\text{front}$  的值分别为( )。

A. 1 和 5;                      B. 2 和 4;                      C. 4 和 2;                      D. 5 和 1。

8. 最大容量为  $n$  的循环队列, 队尾指针是  $\text{rear}$ , 对头指针是  $\text{front}$ , 则队空的条件是( )。

A.  $(\text{rear}+1) \bmod n == \text{front};$                       B.  $\text{rear} == \text{front};$

C.  $\text{rear}+1 == \text{front};$                       D.  $(\text{rear}-1) \bmod n == \text{front}。$

9. 循环队列用数组  $A[0 \cdots m-1]$  存放其元素值, 已知其队头指针  $\text{front}$  指向队头元素, 队尾指针  $\text{rear}$  指向队尾元素, 则当前队列的元素个数是( )。

A.  $(\text{rear}-\text{front}+m) \bmod m;$                       B.  $\text{rear}-\text{front}+1;$

C.  $(\text{rear}-\text{front}+m+1) \bmod m;$                       D.  $(\text{rear}-\text{front}+m-1) \bmod m。$

10. 栈和队列的共同特点是( )。

A. 只允许在端点之外插入和删除元素;    B. 均先进后出;    C. 均先进先出;    D. 无共同点。

11. 用链表方式存储的队列, 插入运算时( )。

A. 仅修改头指针;    B. 头、尾指针都改;

C. 仅修改尾指针; D. 头尾指针可能都需要修改。

12. 下列选项, 哪一个是非线性结构( )。

A. 队列; B. 栈; C. 线性表; D. 二叉树。

13. 树最适合用来表示( )。

A. 有序数据元素; B. 无序数据元素;  
C. 元素之间有父子层次关系的数据; D. 元素之间无联系的数据。

14. 二叉树的第  $K$  层结点数最多为( )。

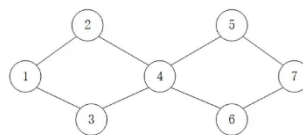
A.  $2^{k-1}$ ; B.  $2^{k+1}$ ; C.  $2^k-1$ ; D.  $2^k+1$ 。

15. 如果森林  $F$  采用“儿子-兄弟”表示法对应的二叉树时 16 个结点的完全二叉树, 问: 森林  $F$  中树的数目和最大树的结点数分别是多少( )。

A. 2 和 8; B. 2 和 9; C. 4 和 8; D. 4 和 9。

16. 对图 1 所示的无向图, 从顶点 1 开始进行深度优先遍历, 可得到顶点访问序列( )。

A. 1, 2, 4, 3, 5, 7, 6; B. 1, 2, 4, 3, 5, 6, 7;  
C. 1, 2, 4, 5, 6, 3, 7; D. 1, 2, 3, 4, 5, 7, 6。



17. 有 6 个节点的无向图, 至少有( )条边才能确定一个连通图。

A. 5; B. 6; C. 7; D. 8。

18. 若将存有 18 个元素的有序表存放在一维数组  $A[19]$  中, 第一个元素存放在  $A[1]$  中, 进行二分查找, 当找到  $A[3]$  的时候, 查找顺序为( )。

A. 1, 2, 3; B. 9, 5, 2, 3; C. 9, 5, 3; D. 9, 4, 2, 3。

19. 对  $n$  个记录的文件进行快速排序, 所需存储空间大致为( )。

A.  $O(1)$ ; B.  $O(n)$ ; C.  $O(\log_2 n)$ ; D.  $O(n^2)$ 。

20. 二维数组  $A[m][n]$  中,  $A[0][0]$  的存放位置是在 644(10),  $A[2][2]$  在 676(10), 每个元素占一个空间, 请问  $A[3][3]$  在( ), 其中脚注(10)表示十进制。

A. 688; B. 678; C. 692; D. 686。

## 二、综合应用题

1. 已知二叉树的先序遍历序列为 ABCDEFG, 中序遍历序列为 CBEFDGA, 试画出此二叉树, 并给出后序遍历序列。

2. 用 C 语言写一个函数, 实现简单选择排序算法, 并根据算法给出下列关键字序列 14, 4, 76, 27, 22, 98, 11 按从小到大的排序过程。

3. 已知有 8 个小伙伴, zhao, qian, sun, li, zhou, wu, zheng, wang, 按照字典的顺序构造这 8 个人的二叉排序树, 并给出查找 wu 所需的步数, 要查找 ma, 要几步才知道 ma 不在序列中。

4. 假设有一个单向循环链表, 结点有三个域: pre, data 和 next, 其中 data 为数据域, next 为指针域, 其值为后继结点的地址, pre 也为指针域, 它的值为空 (NULL), 试编写算法将此链表改为双向循环链表。

5. 设计一个递归算法, 删除不带头结点的单链表  $L$  中所有值为  $x$  的结点。

# 启航计算机 VIP 班习题课数据结构测试答案

## 一、选择题

1. B; 2.D; 3.C; 4.C; 5.C; 6.D; 7.B; 8.B; 9.C; 10.A; 11.D; 12.D; 13.C; 14.D; 15.D;  
16.A; 17.A; 18.D; 19.C; 20.C;

## 二、综合应用题

1. 后续遍历序列: CFEGDBA;

2. 初始序列: 14,4,76,27,22,98,11

第一趟后: 4,14,76,27,22,98,11

第二趟后: 4,11,76,27,22,98,14

第三趟后: 4,11,14,27,22,98,76

第四趟后: 4,11,14,22,27,98,76

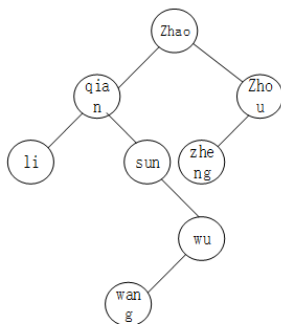
第五趟后: 4,11,14,22,27,98,76

第六趟后: 4,11,14,22,27,76,98

算法代码:

```
#include<stdio.h>
#define N 7
int main(void) {
    int a[N] = {14, 4, 76, 27, 22, 98, 11};
    /*选择排序*/
    for(int i=n-1; i>0; i--) {
        int max_idx=0;
        for(int j=1; j<=i; j++) {
            if(nu[j]>nu[max_idx])
                max_idx=j;
        }
        if(max_idx != i) {
            int temp=nu[i];
            nu[i]=nu[max_idx];
            nu[max_idx]=temp;
        }
    }
}
```

3. 查找 wu 需要 4 步, 查找 ma, 要查找 4 步才知道 ma 不在序列中。



4. 算法代码：

```
#include <stdio.h>
/* 链表元素类型定义 */
typedef int ElemType; // 双循环链表结构体
typedef struct LNode {
    ElemType data;
    struct LNode* next;
    struct LNode* pre;
} LNode, *LinkList;
// 将单向循环链表修改为双向循环链表
int Algo(LinkList L) {
    LinkList r;
    if(L==NULL)
        return -1;
    for(r = L; r->next != L; r = r->next)
        r->next->pre = r;
    L->pre = r;
    return 0;
}
```

5. 算法代码：

```
#include <stdio.h>
/* 链表元素类型定义 */
typedef int ElemType; // 单链表结构体
typedef struct LNode {
    ElemType data;
    struct LNode* next;
} LNode, *LinkList;
void delete_linklist(LinkList L, Elemtype x) {
    if(L != NULL) {
        if(L->data == x) {
            LinkList temp=L;
            L=L->next;
            free(temp);
            delete_linklist(L, x);
        }
        else
            delete_linklist(L->next, x);
    }
}
```