

题号: 423

## 大连海事大学 2003 年研究生招生试题

科目: 数据结构

适用专业: 计算机应用技术

一. 判断下列叙述是否正确: (共 20 分, 每小题 1 分)

- A) ☐ 对任何序列进行排序, 快速排序都比简单插入排序快。
- B) ☐ 每一个网 (带权图) 的最小生成树都是唯一的。
- C) ☐ 同一图的深度遍历序列与广度遍历序列一定不相同。
- D) ☐ 平衡二叉树中所有结点的平衡因子都为 1。
- E) ☐ 算法的优劣与所用计算机无关。
- F) ☐ 输入非法数据不会使健壮的算法出现莫名其妙的状态。
- G) ☐  $n$  个结点的二叉排序树的深度一定小于  $n$ 。
- H) ☐ 栈和队列都是操作受限制的线性结构。
- I) ☐ 将一棵非空树转换成二叉树后, 根结点没有右子树;
- J) ☐ B\_树是有序树。
- K) ☐ 在完成外排序的过程中, 每个记录的 I/O 次数必定相等。
- L) ☐ 能完全拓扑排序的有向图不存在环路。
- M) ☐ 有向图的邻接矩阵一定是不对称的。
- N) ☐ 进行外排序的速度主要取决于所选用的内排序算法的速度。
- O) ☐ 在AoE图中, 只要缩短关键路径上某个活动的时间, 就可以缩短整个工程的时间。
- P) ☐ 给定  $n$  个权值不相同的结点, 则以这  $n$  个结点作为叶子构造出的Huffman树的结构是唯一的。
- Q) ☐ 在线索二叉树中, 树中任何一个结点在相应遍历序列中的直接前驱都可以根据线索找到。
- R) ☐ 在线性表的顺序存储结构中, 删除任何一个数据元素都必须移动表中的其它数据元素。
- S) ☐ 希尔排序采用了多趟插入排序, 因此常常比简单插入排序花费的时间多。
- T) ☐ 在索引顺序表中分块查找时, 其平均查找长度不仅与表中分块数有关, 还与每块中元素个数有关。



二. 请选择准确的字或词填入空缺位置, 构成正确完整的描述。(共 30 分, 每小题 2 分)

- 1) 栈结构可以实现“\_\_\_\_\_进\_\_\_\_\_出”的功能。
- 2) 提高外排序速度的核心工作是减少记录的\_\_\_\_\_次数。
- 3) 在 15 个记录折半查找中, 最多的比较次数是\_\_\_\_\_。
- 4) 具有  $n$  个顶点的简单无向连通图, 其边数至少为\_\_\_\_\_。
- 5) 具有  $n$  个叶子结点的 Huffman 树中, 非叶子结点的数目是\_\_\_\_\_。
- 6) 对  $n$  个记录的简单插入排序, 在最好情况下的时间复杂度为  $O(\text{_____})$ 。
- 7)  $n$  个结点的二叉树的二叉链表存储结构中, 一定有\_\_\_\_\_个空链域。
- 8) 设  $T$  是二叉树, 且  $T$  的后序遍历序列与先序遍历序列相同, 则  $T$  的结点数最多为\_\_\_\_\_个。
- 9) 对一棵排序二叉树的所有结点进行\_\_\_\_\_遍历, 所得到的遍历序列是有序的
- 10) 设  $T$  是一棵完全二叉树, 则  $T$  的根结点的左子树的结点数  $N_l$  与右子树的结点数  $N_r$  相比, 其大小关系是\_\_\_\_\_。
- 11) 对广义表  $A = (a, (c, (a, b), d))$  实施运算  $\text{Tail}(\text{Head}(\text{Tail}(A)))$  后的结果是\_\_\_\_\_。
- 12) 设  $S = 'xyyyzxyyyxyy'$ ,  $T = 'xy'$ , 则用  $T$  替代  $S$  中的  $'xyy'$  后,  $S$  将变换为\_\_\_\_\_。
- 13)  $B$ \_树是一种平衡多路查找树, 7-阶  $B$ \_树的结点中的关键字数至多有\_\_\_\_\_个, 最少有\_\_\_\_\_个。
- 14) 设循环队列存放在一维数组  $\text{sq.data}[0..m]$  中, 则插入一个记录后, 队尾指针  $\text{sq.rear}$  在循环意义下的加工操作可表示为\_\_\_\_\_。
- 15) 设二维数组  $B[1..4, 1..3]$  以行主序顺序存储在一个连续的存储空间内, 每一个数据元素占一个存储单元, 且  $B[1, 1]$  的址是 10, 则  $B[3, 2]$  的存储地址是\_\_\_\_\_。





## 八. 请写出下列算法的输出 (10 分)

```

Void main() { // EnQ( Q, x)表示入队操作; DeQ( Q, x)表示出队操作
    Queue Q;
    char x='a', y='b';

    InitQueue;

    EnQ( Q, 'c');      EnQ( Q, 'g');
    EnQ( Q, x);        DeQ( Q, y);
    EnQ( Q, 'k');      DeQ( Q, x);
    EnQ( Q, y);        EnQ( Q, 'r');
    while( ! QueueEmpty( Q ) ) {
        DeQ( Q, y);
        Printf( y );
    }
    printf( x );

} // End of main

```

## 九. 请写出递归算法 findIt ( 10 ) 执行过程中的所有输出 (10 分)

```

int findIt( int m ) {
    int n;

    if (m <= 0) {
        n = m - 2 ;
    } else {
        n = findIt( m - 3 ) + m ;
    }
    printf( n );
    return( n );
}

```

## 十. 编程题 (15 分)

请编程实现: 在带头结点的非空单链表 L 中, 将元素值为 X 的结点全部放到单链表 L 的尾部。