

西安交通大学 2015 年招收攻读硕士学位研究生入学考试试题答案

科目：820（现在改为 915，考试科目一样，代码变了而已）

一、 选择题

1. D 数据结构是一门研究非数值计算程序设计问题的数据映
像以及他们之间关系和运算的学科。2. C 3. A 4. D 待排序序列接近无序，快
速排序效率越高。5. A 6. B 7. C 8. A 深度优先搜索遍历类似于二叉树的先序
遍历，广度优先搜索遍历类似于树的层次遍历。9. B 10. D

二、 判断题

1. X 无向图是连通的，有向图的一个顶点到其余顶点都有边。
2. √ 3. X 当无向连通图存在权值相同的多条边时，最小生成树可能不
唯一。4. X 5. √
6. X 直接插入排序在最好情况下的时间复杂度为 $O(n)$
7. X 冲突时不可避免的，与装填因子有关。冲突产生概率与装填
因子的大小成正比。散列查找成功的平均查找长度与装填因子有
关。
8. X 9. X 有序表，顺序存储。
10. X 按行优先，三行四列为第 24 个数。按列优先，四行三列为第 16 个数。

三、 简答题

1. 由某个集合上的一个偏序关系得到整个集合上的一个全序，这个操作称为拓
扑排序。应用场景：应用于具有先后顺序的场景中，判断图中是否有环。
2. n 个结点，构成 $C(n, 2n) = 2n!/n+1$ 种不同的二叉树。
3. 哈夫曼树：带权路径长度 WPL 最小的二叉树。
哈夫曼编码：对构造好的哈夫曼树，约定左分支表示 0，右分支表示 1。从根结
点到叶子结点的路径上的编码组成该叶子结点的赫夫曼编码。
哈夫曼树，只有度为 0 和度为 2 的结点。 n 个叶子结点， $n-1$ 个度为 2 的结
点，一共 $2n-1$ 个结点。度为 m 的哈夫曼树，叶子结点为 n ，则非叶子结点数
位 $\lceil (n-1)/(m-1) \rceil$ 。

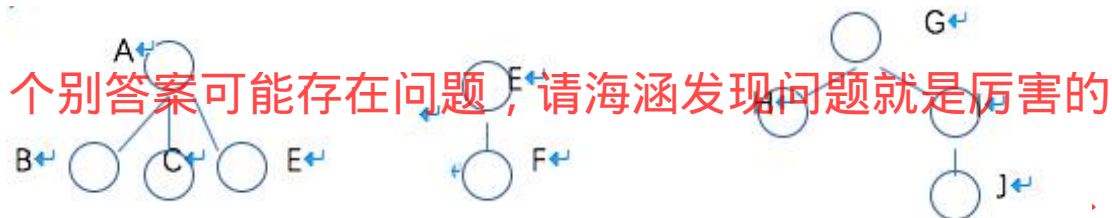
四、 解答题

1. 相关的答案在高分笔记第二章队列的相关操作 ①判空： $Q.rear == Q.front$ 判
满： $(Q.rear+1)\%10 == Q.front$
②进队： $Q.rear = (Q.rear+1)\%10$; $Q.data[Q.rear] = x$;
出队： $Q.front = (Q.front+1)\%10$; $x = Q.data[Q.front]$;
③

0	1	2	3	4	5	6	7	8	9
A	B	C	H	I(rear)					front

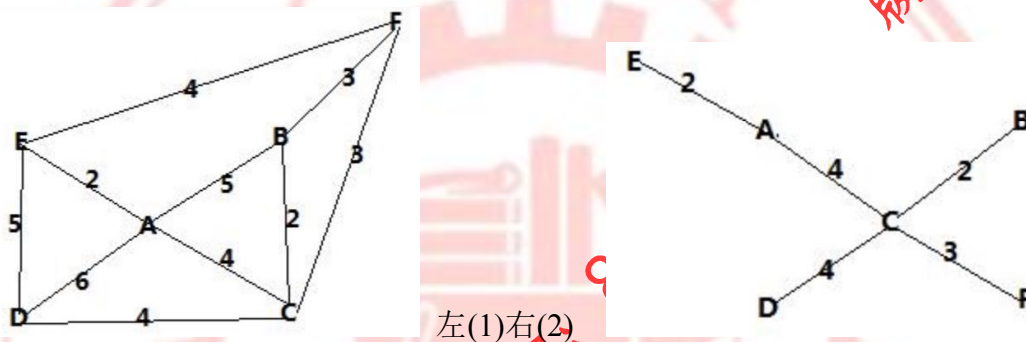
2. ①中序：BCDAFEHJIG 后序：DCBFJIHGEA

②



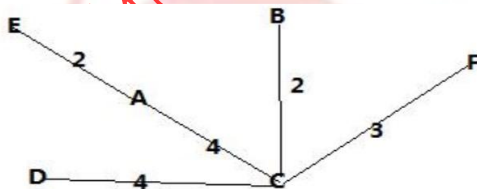
③森林的前序：ABCDEFGHJIJ；中序：BCDAFEHJIG

3.(1)



②由于给定了存储结构，则访问的先后顺序就确定了。CADBFE

③



4. ①选择排序的算法代码，具体代码见高分笔记书上例题，必会题。

②13 22 10 18 35 26 35 28

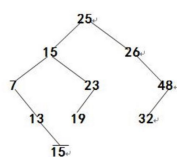
13 22 10 18 35 26 35 28

10 13 22 18 35 26 35 28

10 13 18 22 35 26 35 28

③基数排序题很简单，按照高分笔记的步骤正常做就行，送分题。

5. ①



其中，对于同一个关键字，则不进行插入，只进行判断，所以最后一个 15 是只进行判断，不做插入，所以图上的 15 做了标记。

②

地址	0	1	2	3	4	5	6	7	8	9	10
值	48	32	15	23	7	13	19	25		15	26
比较次数	3	3	5	1	1	1	1	1		1	2

③ $ASL1 = (1 + 2 \times 2 + 3 \times 3 + 4 \times 3 + 5) / 10 = 3.1$

个别答案可能存在问题，请海涵发现问题就是厉害的

五、编程题

1.(题库原题，自己找下，给一个参考答案)

```
#include <stdio.h>
```

```
void add( int s[100][100], int n )
```

```
{
```

```
    int i, j, sum1 = 0, sum2 = 0;
```

```
    for ( i = 0; i < n; i++ )
```

```
        for ( j = 0; j < n; j++ )
```

```
            if ( i == j )
```

```
                sum1 += s[i][j];
```

```
    for ( i = 0; i < n; i++ )
```

```
        for ( j = n - 1; j >= 0; j-- )
```

```
            if ( (i + j) == (n - 1) )
```

```
                sum2 += s[i][j];
```

```
    printf( "主对角线之和为 : %d\n", sum1 );
```

```
    printf( "副对角线之和为 : %d\n", sum2 );
```

```
}
```

```
void main()
```

```
{
```

```
    int n, i, j;
```

```
    int s[100][100];
```

```
    printf( "请输入矩阵的阶数: n" );
```

```
    scanf( "%d", &n );
```

```
    printf( "请输入矩阵: \n" );
```

```
    for ( i = 0; i < n; i++ )
```

```
        for ( j = 0; j < n; j++ )
```

```
scanf( "%d", &s[i][j] );  
add( s, n );  
}
```

2、题库原题（见本资料题库第 168 页第 4 题和第 5 题）下面答案是更简化的一个答案。

个别答案可能存在问题，请海涵发现问题就是厉害的

```
#include <stdio.h>  
void main()  
{  
    int i, n, num = 0, let = 0, oth = 0; char s[80];  
    gets( s ); n = strlen( s );  
    for ( i = 0; i < n; i++ )  
        if ( s[i] >= '0' && s[i] <= '9' )  
            num++;  
        else if ( s[i] >= 'a' && s[i] <= 'z' || s[i] >= 'A' && s[i] <= 'Z' )  
            let++;  
        else oth++;  
    printf( "数字: %4d  频率: %.2f\n 字母: %4d  频率: %.2f\n 其它: %4d  频率: %.2f", num, num * 1.0 / n, let, let * 1.0 / n, oth, oth * 1.0 / n );  
}
```

3.（题库原题，在本资料 191 页第 6 题，这个编程题在 2014 年真题中也是原题）

4.（这个当时直播课我们给学生画过单链表的这个题）

```
int caortx( struct LNode *HL, int x )  
{  
    int k = 0;  
    LNode *p = HL —> next; /* 设 HL 有头结点 */  
    while ( p != NULL )  
    {  
        if ( p —> data == x )  
            k++;  
        p = p —> next;  
    }  
    return(k);  
}
```

西交软件科创团队创作：87172978