

考试科目: 数据结构

得分: _____

专业: _____

姓名: _____

学号: _____

(请在试卷及每张答卷上写上姓名与学号)

一、单项选择题 (2分×10=20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. $O(\log_2 n)$ C. $O(n)$ D. $O(n^2)$
2. 设指针 q 指向单链表中结点 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 插入结点 X 的操作序列为()。
A. s->next=p->next; p->next=s; B. q->next=s; s->next=p;
C. p->next=s->next; s->next=p; D. p->next=s; s->next=q;
3. 若元素 a、b、c、d、e、f 依次进栈, 允许进栈、退栈操作交替进行, 但不允许连续三次进行退栈操作, 则不可能得到的出栈序列是()。
A. d c e b f a B. c b d a e f C. b c a e f d D. a f e d c b
4. 设二维数组 A[1..m,1..n]按行存储在一维数组 B[1..mn]中, 则二维数组元素 A[i,j] 在一维数组 B 中的下标为()。
A. $n*(i-1)+j$ B. $n*(i-1)+j-1$ C. $i*(j-1)$ D. $j*m+i-1$
5. 下列关于最小生成树的叙述中, 正确的是()。
 - I. 最小生成树的代价唯一
 - II. 所有权值最小的边一定会出现在所有的最小生成树中
 - III. 使用普里姆算法从不同顶点开始得到的最小生成树一定相同
 - IV. 使用普里姆算法和克鲁斯卡尔算法得到的最小生成树总不相同
A. 仅 I B. 仅 II C. 仅 I、III D. 仅 II、IV
6. 5个字符有如下4种编码方案, 不是前缀编码的是()。
A. 01, 0000, 0001, 001, 1 B. 011, 000, 001, 010, 1
C. 000, 001, 010, 011, 100 D. 0, 100, 110, 1110, 1100
7. 对有 n 个结点、e 条边且使用邻接表存储的有向图进行广度优先遍历, 其算法时间复杂度是()。
A. $O(n)$ B. $O(e)$ C. $O(n+e)$ D. $O(n^2)$
8. 设有序表中的元素为(13, 18, 24, 35, 47, 50, 62), 则在其中利用二分法查找值为 24 的元素需要经过()次比较。

- A. 1 B. 2 C. 3 D. 4
9. 设有向无环图 G 中的有向边集合 $E = \{<1, 2>, <2, 3>, <3, 4>, <1, 4>\}$, 则下列属于该有向图 G 的一种拓扑排序序列的是 ()
 A. 1, 2, 3, 4 B. 2, 3, 4, 1 C. 1, 4, 2, 3 D. 1, 2, 4, 3
10. 设有一组初始记录关键字序列为(34, 76, 45, 18, 26, 54, 92), 则由这组记录关键字生成的二叉排序树的深度为 ()。
 A. 4 B. 5 C. 6 D. 7

二、填空题(2 分×10=20 分)

- _____是一种限定仅在表尾进行插入或删除操作的线性表, 其运算遵循后进先出的原则。
- 设顺序循环队列 $Q[0..M-1]$ 的头指针和尾指针分别为 F 和 R, 头指针 F 总是指向队头元素的前一位置, 尾指针 R 总是指向队尾元素的当前位置, 则该循环队列中的元素个数为 _____。
- 广义表 ((a), ((b), c), (((d)))) 的深度是 _____。
- 一棵高度为 5 的完全二叉树中的结点数至少为 _____ 个。
- 设一棵二叉树的先序遍历序列为 ABDGHCEFI, 中序遍历序列为 GDHBAECI F, 则这棵二叉树的后序遍历序列为 _____。
- 含有 n 个节点的二叉链表(不带头结点)中总共有 _____ 个空指针, 若在其上添加中序遍历线索, 则线索化后得到的中序线索链表中含有 _____ 个空指针。
- 若一棵哈夫曼树的结点总数为 2001 个, 则它有 _____ 叶子结点。
- 已知图 G 的邻接表如图 1 所示, 其从顶点 v1 出发的深度优先搜索序列为 _____, 广度优先搜索序列为 _____。

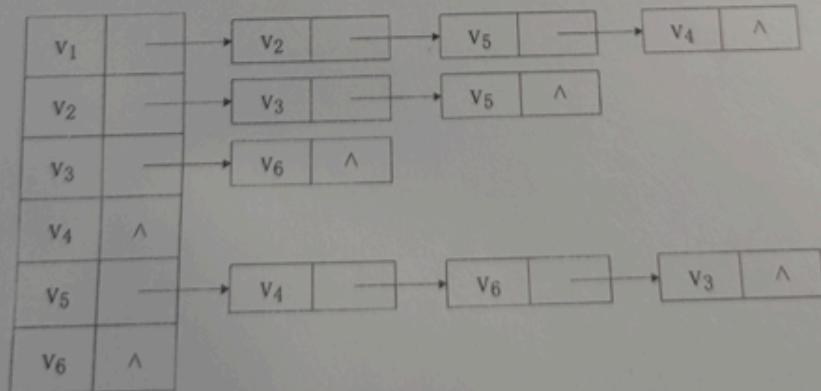


图 1

三、应用题(10 分×3=30 分)

- 有一份电文中共使用 8 个字符: a、b、c、d、e、f、o、i, 它们的出现频数依

次为 10, 20, 15, 32, 40, 60, 26, 18。请画出相应的哈夫曼树，并给出每个字符的哈夫曼编码。

2. 使用普里姆 (Prim) 算法构造出如图 2 所示的无向图 G 的一棵最小生成树。

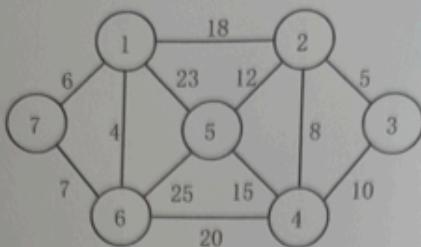


图 2 无向图 G

3. 已知图 3 所示的一个 3-阶 B 树，请画出插入关键字 g 后的 B 树，然后在此基础上删除关键字 l，请画出删除关键字 l 后的 B 树。

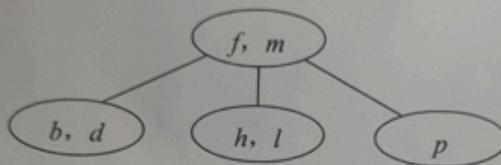


图 3 3-阶 B 树

四、算法设计题(15 分×2=30 分)

1. 给定连通的无向无权图 G，打印输出两个不同顶点 s 和 t 之间所有简单路径的顶点序列（路径上顶点不重复）。假设图的存储结构为邻接矩阵，如下：

```
#define MaxVertexNum 30 //最大顶点数
```

```
typedef char VertexType; //顶点数据类型为字符型,s 和 t 是简单路径的起点和  
//终点, 假定任意两个顶点的数据都不同
```

```
typedef int EdgeType; //1: 表示有边存在; 0: 表示两个顶点间无边
```

```
typedef struct{
```

```
    VertexType vexs[MaxVertexNum] //顶点表
```

```
    EdgeType edges[MaxVertexNum][MaxVertexNum]; //邻接矩阵
```

```
    int n, e; //图中当前的顶点数和边数
```

```
} MGraph;
```

```
MGraph g; //假定图的顶点和边信息都已经存储在 g 中, 可直接对 g 进行操作  
或运算。
```

2. 两有序数组中寻找中位数：假设有两个数组 A 和 B，分别包含有 m 和 n 个整数，数组元素均按照递增顺序排列，请设计有效的算法确定两个数组中所有元素的中位数（“有效”指算法的时间复杂度尽可能的低，“中位数”指集合中大小居中的那个数），并给出相应的时间复杂度（提示：最优的算法时间复杂度可达 $O(\log(m+n))$ ）。例如，算法输入数组 A={1,2,3}，B={3,5}，则算法输出为 3 ({1,2,3,3,5} 的中位数)；若算法输入 A={1,2,3}，B={2,4,5}，则算法输出为 2 或 3 ({1,2,2,3,4,5} 的中位数)。