期末考试降至，将期末考试题型以及一些常见题目列举出来，希望大家抓紧时间复习，取得一个好成绩！

# 考试题型

一、简答题（4小题，20分）

二、选择题（10小题，10分）

三、填空题（10小题，20分）

四、构造题（4小题，30分）

五、算法设计题（2小题，20分）

# 典型题目

**注意：**以下仅是参考，**不是考试题目，希望别被误导☺☺☺**。考试范围覆盖全书，但以6、7、8、9章为主。

***一、简答题***

1. 四种基本逻辑结构及其图示。
2. 抽象数据类型的定义。
3. 线性表顺序存储与链式存储的优缺点。
4. 一维数组和顺序表的异同。
5. 栈、队列和线性表的异同
6. 一棵有n个结点的k叉树采用k叉链表存储，则空链域的数目是多少？写出求解过程。
7. 在图的遍历中，设置访问标志数组的作用是什么？
8. 求最小生成树有哪些方法？各自适合什么类型的图。
9. 什么是关键路径？
10. 折半查找的前提有哪些？
11. 什么是平衡二叉排序树？平衡因子的取值范围有哪些？
12. 分块查找的基本思想。
13. 分析二叉排序树的查找性能。（最好、最坏和平均查找性能）
14. 设有5000个待排序的记录关键字，如果需要用最快的方法选出其中最小的10个记录关键字，则快速排序、堆排序、归并排序、直接插入排序中，哪些排序方法最高效，并说明理由。
15. 什么是排序的稳定性？列举至少个稳定的排序算法和三个不稳定排序算法。
16. 分析直接插入排序（冒泡、简单选择）的最好性能和最坏性能。（比较和移动次数）

***二、选择题（仅供参考）***

1、在一棵平衡二叉排序树中，每个结点的平衡因子的取值范围是（ ）

A．-1~1 B．-2~2 C．1~2 D．0~1

2、对于顺序表，其算法时间复杂度为O(1)的运算是（ ）

A）将n个元素从小到大排序

B）从线性表中删除第i个元素(1≤i≤n)

C）查找第i个元素(1≤i≤n)

D) 在第i个元素之后插入一个新元素(1≤i≤n)

3、 在单链表中，下列说法正确的是（）

A.单链表中头结点是必不可少的； B.单链表中头指针是必不可少的；

C.在单链表中可以实现随机存取； D.单链表的存储密度小于顺序表

4、假设以数组A[M]存放循环队列的元素，其头尾指示器分别为front和rear，则当前队列中的元素个数为( )。

A.rear-front+1 B. (rear-front+M)%M

C. (front - rear +M)%M D. (rear-front+M)%M

5、已知广义表L=((a,b,c),(d,e,f)),运用下列（）运算可以得到结果：e。

A.head（tail(L)）　　　 B.tail(head(L))

C.head(tail(head(tail(L))))　D.head(tail(tail(head(L))))

6、 线索二叉树中，某结点p是叶子结点,下列（）表达式的值为真。

A.p->lchild= =NULL B.p->ltag= =1&&p->rtag= =1

C.p->ltag= =0 D.p->lchild= =NULL&& p->rchild= =NULL

7、一个具有567个结点的完全二叉树的高度为（ ）

A.9 B.10 C.11 D.12

8、具有n个顶点的强连通图，至少有（ ）条边

A．n-1 B． n C．n(n-1) D．n(n-1)/2

9、下列 ( )的邻接矩阵一定是对称矩阵。

A．有向图 B.无向图 C.AOV网 D.AOE网

10、已知哈希函数为H(key)=key%p,哈希表长度为m,采用开放定址法处理冲突，若H(k1)地址冲突，则处理冲突定址为（）。

A．(H(k1)+di)%p B. H(k1)+di C. (H(k1)+di)%m D. H(k1)-di

11、快速排序方法在（ ）情况下最不利于发挥其长处（性能最差）。

A．要排序的的数据量太大 B．要排序的数据中含有多个相同的值

C．要排序的数据有序 D．要排序的数据个数是奇数

12、己知有序表为(13,19,24,33,47,57,64)，当用二分法查找64时，需 （ ）次查找成功。

A）1 B）2 C）3 D）4

13、100个结点的完全二叉树采用顺序存储，从1开始编号，则编号最大的非叶子结点的编号应该是（ ）。

A) 100 B) 49 C) 50 D) 51

***三、填空题 （仅供参考）***

1、设有数组A[1..8][1..10],每个元素长度为3字节，数组从内存首地址SA开始顺序存放，当以**列为主存**放时，元素A[5][8]的存储首地址为( )。

2、88个结点的完全二叉树，其高度为（ ）。

3、长度为n的有序单链表，若查找每个元素的概率相等，则查找成功的平均查找长度为（ ）

4、对任意一棵有n个结点的树，这n个结点的度之和为（ ）。

5、n个顶点构成一个环，该无向图共有（ ）棵生成树。

6、n个顶点的无向完全图，共（ ）条边

7、设森林F中有三棵树，第一、第二和第三棵树的结点个数分别为M1、M2和M3。与森林F对应的二叉树根结点的左子树上的结点个数是：（ ）

8、已知一棵完全二叉树第7层有10个叶子结点则该二叉树中叶子结点总数为（ ）。37或者235个

9、折半查找的两个前提条件分别是（ ）和（ ）

10、下面的算法，s语句的语句频度是（ ）。

for(int i=1;i<n-1;i++)

for(int j=n;j>=i;j--)

s;

11、 有向图的邻接表表示法中，第i条链上边表结点的个数为该顶点的（ ）。

12、已知一个带头结点的链栈，其头指针为top；指针s指向一个新结点，要将结点s进栈，则进栈的语句应为：（ ）和（ ）。

13、有一种排序算法，其时间复杂度为O（n2）,关键字比较次数与待排序记录的初始排列顺序无关且排序不稳定，则该排序算法是（ ）

14、一棵具有10个叶子结点的哈夫曼树中分支结点（即非叶子结点）数为 （ ）**。**

15、{10，12，13，7，8，9，30，4，5}采用快速排序方法，以第一个元素为中轴元素，一趟快速排序后的结果为（ ）。

16、在顺序存储的线性表上R[4]上，从前向后进行顺序查找。若查找第一个元素的概率是1/2，查找后三个元素的概率各为1/6。则查找成功的平均查找长度为（ ）。

***四、构造题***

1、构造二叉树。

（1）欲建立一棵二叉链表存储的二叉树，输入其扩展先序序列ABC##DE#G##F###，画出相应的二叉树。（其中#代表空指针）

（2）某二叉树的后序遍历序列为EDBFCA，中序遍历序列为EDBAFC，请画出该二叉树，并写出该二叉树的顺序存储结构。

（3）二叉树和树、森林之间的相互转换。

树的先根遍历《=》相应二叉树的先序遍历

树的后根遍历《=》相应二叉树的中序遍历

森林的先根遍历《=》相应二叉树的先序遍历

森林的中根遍历《=》相应二叉树的中序遍历

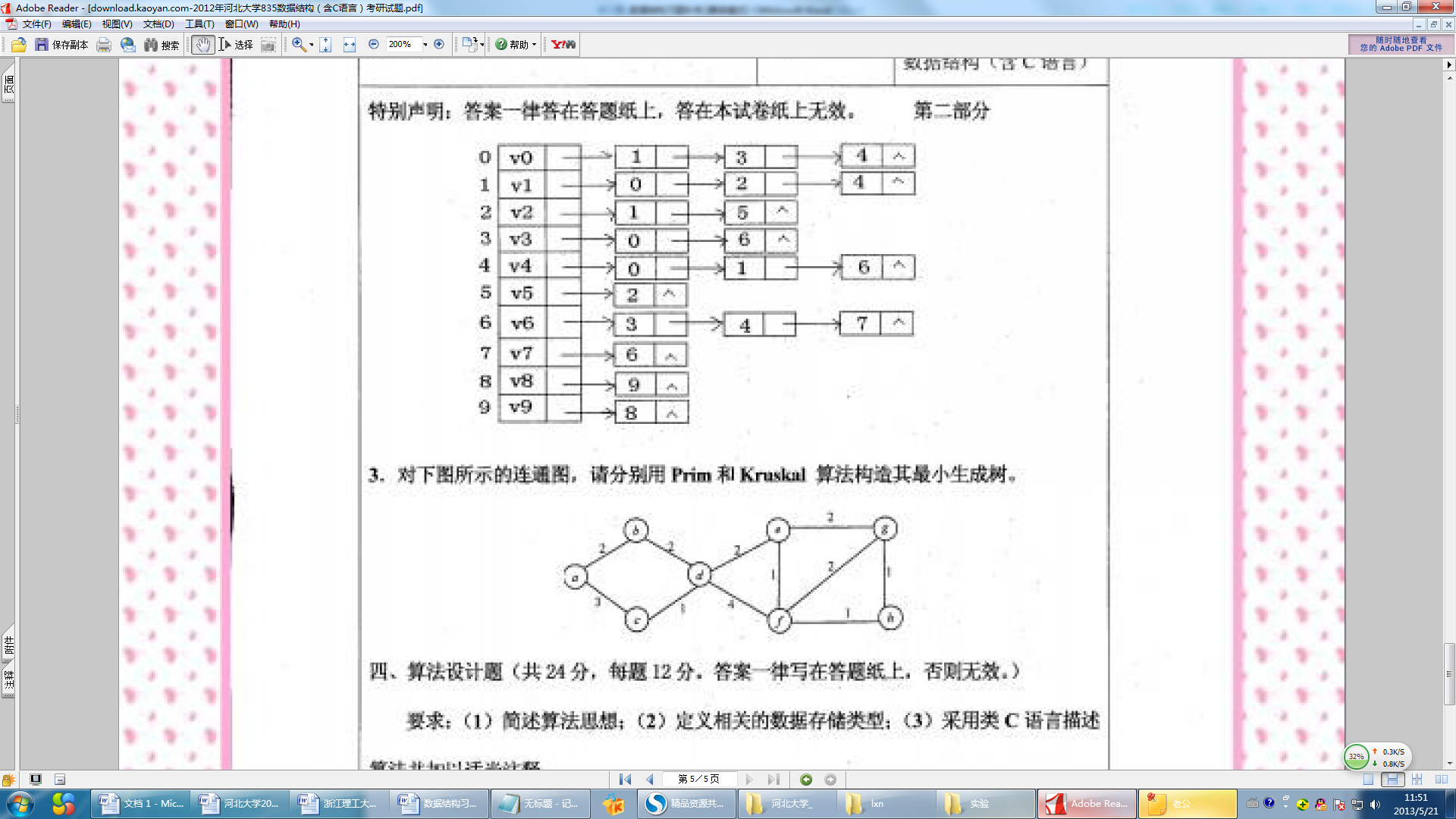
森林的后根遍历《=》相应二叉树的后序遍历

2、哈夫曼编码。假定用于通信的某电文仅由8个字母构成，各字母在电文中出现的频率分别为（12，5，3，7，14，21，9，15）。请完成：

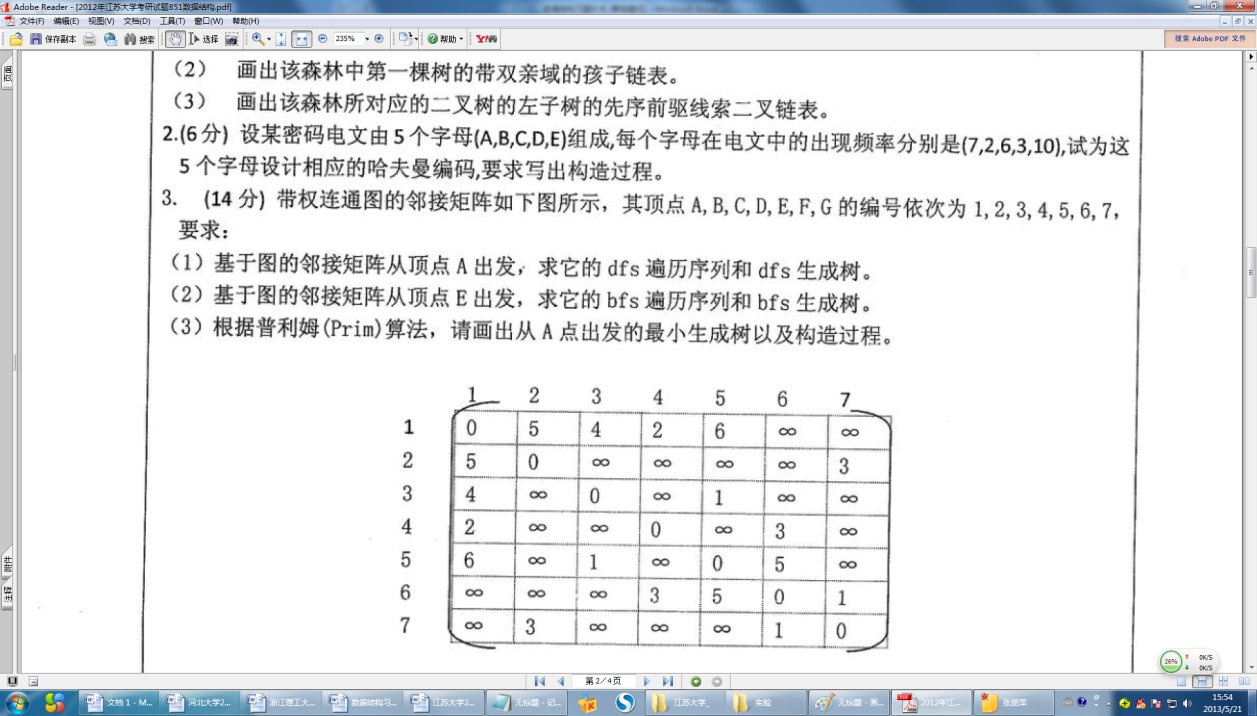
（1）构造哈夫曼树；（2）为这8个字母设计不等长的Huffman编码，并计算WPL。

* 1. 已知一个图的顶点为A、B、C、D，其邻接矩阵的下三角元素全为0（包括主对角线元素），其他元素均为1。请画出该图，并给出其邻接表。

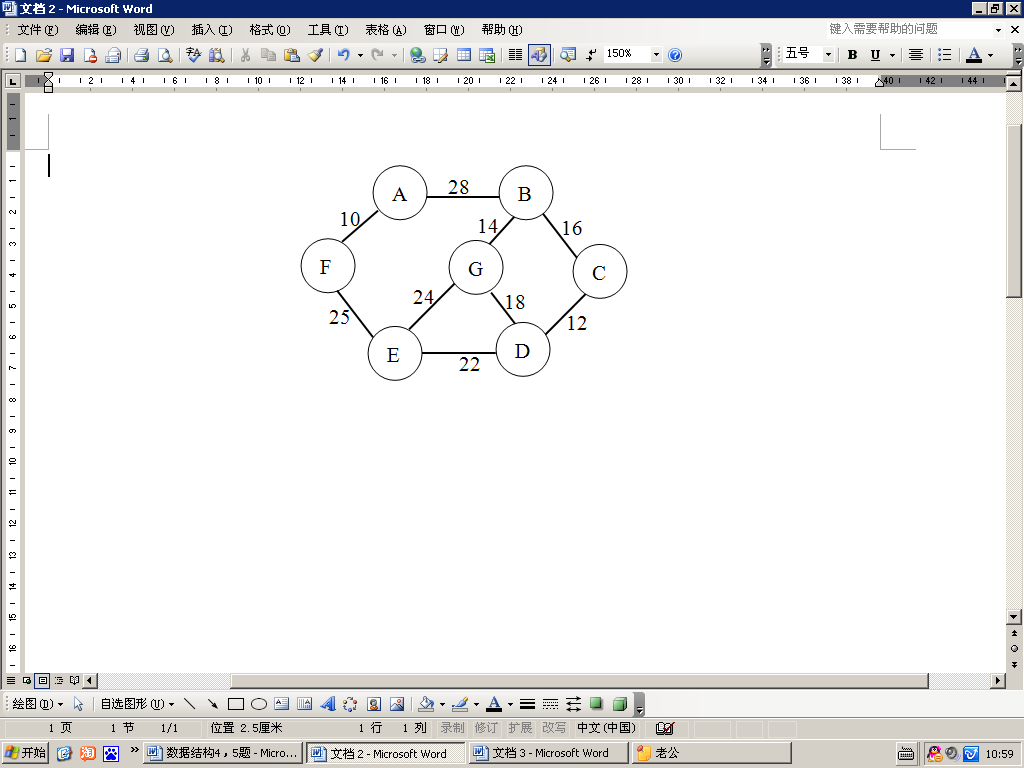
4、某有向图的邻接表存储结构如图所示，请写出从v0出发的深度和广度遍历序列，以及深度和广度优先生成树。



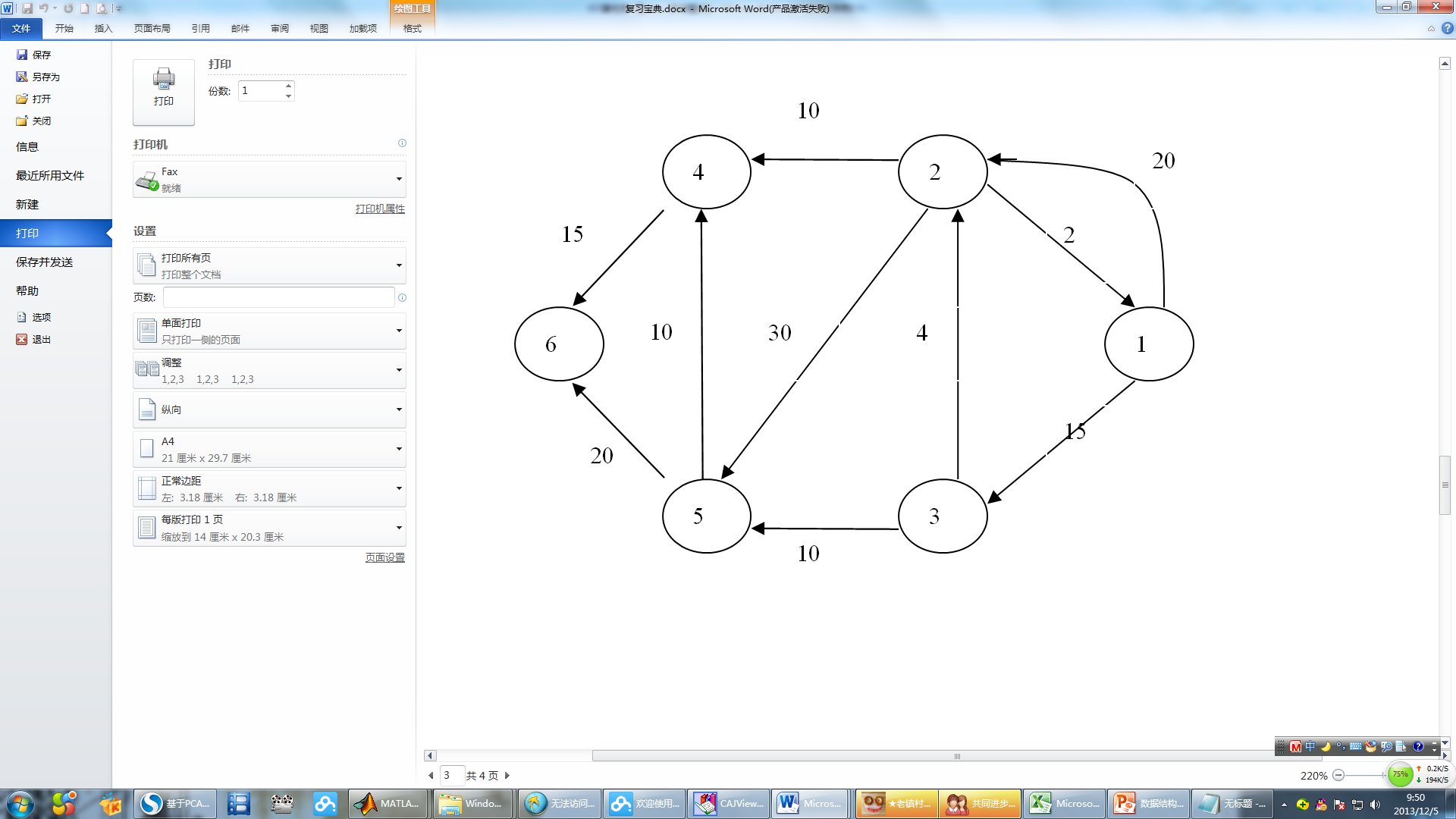
5、图的邻接矩阵存储结构如图所示，请写出从1出发的深度和广度遍历序列，以及深度和广度优先生成树。



6、最小生成树。用普利姆算法（或者克鲁斯卡尔）从顶点A出发，构造如图所示连通网的最小生成树（写出过程）。



7、最短路径。在下图中，求顶点1到其余各点的最短路径及路径长度。



8、折半查找性能分析。画出10个元素的折半判定树，并计算等概率情况下的查找成功与不成功的查找长度。

9、二叉排序树。输入序列{38，25，74，63，52，48}，构建二叉排序树，并计算等概率情况下查找成功的平均查找长度。

10、哈希查找。一个线性序列{38，25，74，63，52，48}，假定采用散列函数Hash(key)=key%7来计算散列地址，将其散列存储在A[0~9]中，采用线性探测再散列解决冲突（或链地址法解决冲突）。构造哈希表，并计算等概率情况下的查找成功和不成功的平均查找长度。

11、各种排序算法。对序列{57，40，38，11，13，34，48，75，6，19，9，7}欲进行递增排序，请写出直接插入排序（或简单选择排序、冒泡排序、快速排序、归并排序）前三趟的排序结果。

12、排序。对序列{57，40，38，11，13，34，48，75，6，19，9，7}，采用堆排序算法进行 ***递减（递增）*** 排序，请构造相应的初始堆即建初堆（只写初堆结果即可）。

***五、算法设计题***

**1、链表。**

（1）一带头结点的单链表，删除最大（最小）元素。

（2）一带头结点的单链表，以第一个元素为基准，将小于该元素的结点全部放到前面，大于该元素的结点全部放到其后。

（3）某个带头结点的有序单链表L（元素从小到大），编写算法插入值为x的元素，使插入后的链表依旧有序。

（4）已知带头结点的单链表L，编写算法删除L中从第k个结点开始的n个结点。

（5）编写算法，求某一元多项式的导数。函数说明为PolyNode\* Derivative(PolyNode \*PL)，参数为一元多项式，返回值为该一元多项式导数的一元多项式链表头指针。

**2、栈与队列**

（1）运用栈的基本运算，编写将一个十进制整数n转换为r(r<=16)进制的算法。

（2）运用栈的基本运算，判断一个表达式的括号是否匹配。

（3）在一个带头结点的循环链队列中只有尾指针rear，请给出这种队列的入队和出队操作的实现过程。

**3、二叉树。**

（1）二叉树采用顺序存储，编写算法实现该二叉树的中序遍历。

（2）二叉树采用链式结构存储，假设二叉树上的结点值各不相同，设计一个在二叉树中求值为e的结点的双亲结点。

（3）二叉树采用链式结构存储，编写算法实现二叉树的层次遍历。

（4）二叉树采用链式结构存储，编写算法，实现二叉树的左右子树交换。

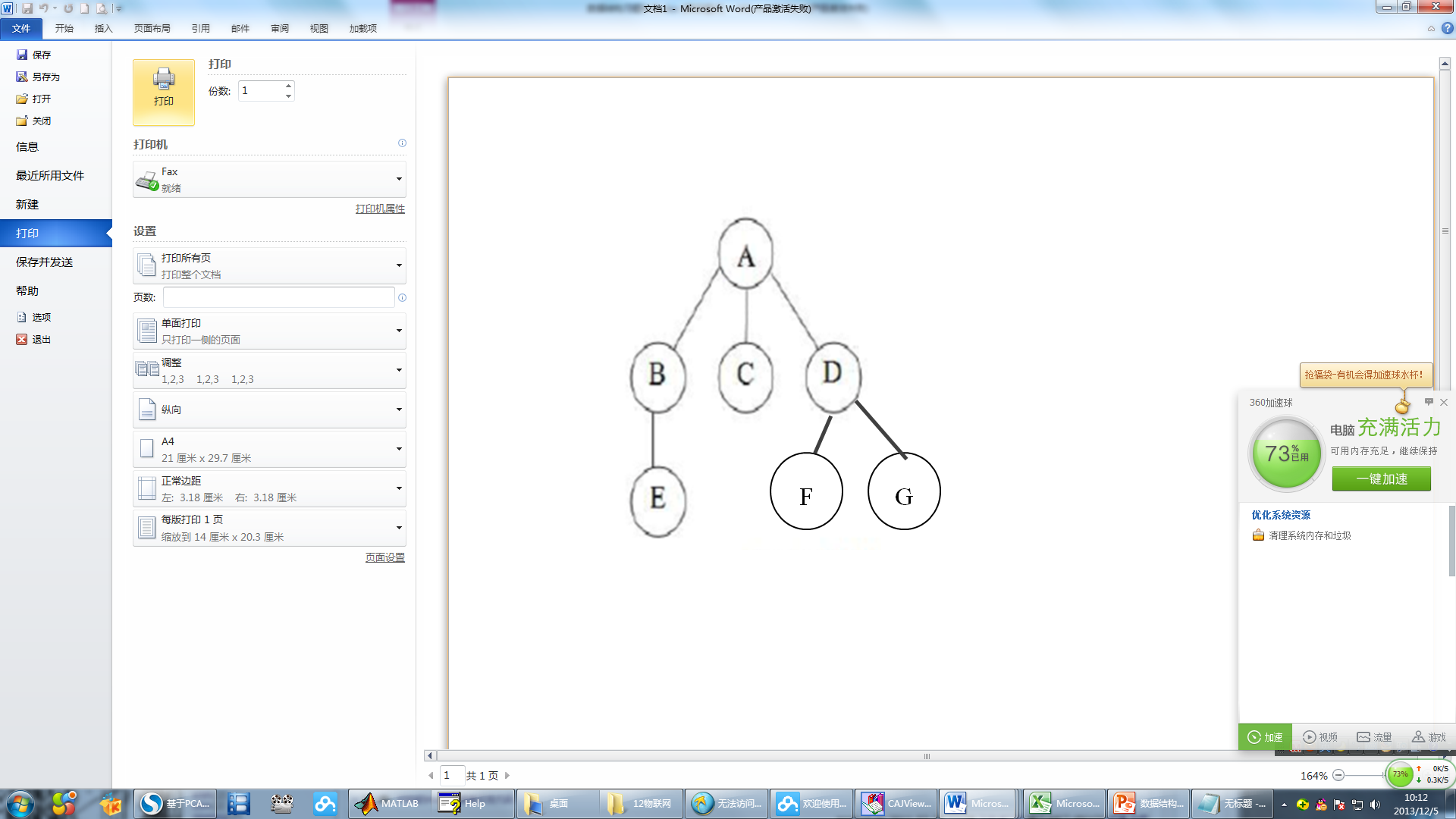
（5）二叉树采用链式结构存储，编写算法，对某二叉树实现其后序线索化。

（6）二叉树采用链式结构存储，编写算法，在某先序线索二叉树中，查找某结点的后继结点。（在后序线索树中找前驱；在中序线索树中找前驱后继。）

（7）某二叉树采用链式结构存储，编写算法，为每个结点设置其所在层次。

（8）某树采用孩子-兄弟链表示，编写算法求该树的高度（或统计叶子的个数）

（9） 某树采用孩子-兄弟链表示，编写算法按（双亲，孩子）格式输出。如图所示。



输出为

(A,B) (A,C) (A,D) (B,E) (D,F) (D,G)

**4、图。**

（1）在图的邻接矩阵存储结构上，实现图的广度（深度）优先遍历。（一般会给出图的存储结构的C语言描述）

（2）在一无向图中，输出其所有连通分量（每个连通分量由字符结点组成，不同连通分量之间用空格或回车分开即可）。

（3）编写算法，在某图中，判断从vi到vj是否有路径相通。

**5、查找。**

（1）编写折半查找算法。

（2）编写算法，构建哈希表。哈希函数未知，处理冲突采用线性探测再散列。

（3）编写算法，在已有哈希表中实现Hash查找。哈希函数未知，处理冲突采用线性探测再散列。

**6、排序。**

（1）对单链表head进行选择排序，排序后结点值从小到大排序。

（2）编写算法实现冒泡排序（直接插入排序、简单选择排序）。