1．假设用于通讯的电文仅由8个字母A、B、C、D、E、F、G、H组成，字母在电文中出现的频率分别为：0.07，0.19，0.02，0.06，0.32，0.03，0.21，0.10。请为这8个字母先构造哈夫曼树，再设计哈夫曼编码。在构造哈夫曼树时，要求每个结点左子树根结点权值小于右子树根结点权值，设计哈夫曼编码时规定向左为0，向后为1。 计算wpl。

2．已知无向图G如下图所示：

（1）求从顶点v1出发的深度优先遍历序列，当某个顶点有多个邻接点时，**编号小的顶点优先访问**；

（2）根据Prim和Kruscal算法，求图G从顶点v0出发的最小生成树，**要求写出详细过程**。



3．已知关键码序列为(53，17，19，61，98，75，79，63，46，49)，将关键字码散列到地址空间为{100，101，102，103，104，105，106，107，108，109}内，若产生冲突采用线性探测再散列法解决，选用的哈希函数为H（key）=key%7+100。构造哈希表，计算等概率下查找成功时的平均查找长度。

4．设待排序序列为{10,18,4,3,6,12,1,9,15,8}，请写出采用希尔排序每一趟的排序结果。增量序列为5，3，2，1。

1. 对长度为15的有序表，给出折半查找判定树，并求等概率情况下的平均查找长度。

6.已知一棵用二叉链表方式存储的二叉树，请用中序遍历方式统计其非叶子结点数。

要求：

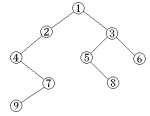
（1）定义二叉树的存储结构

（2）请编写递归函数int SumNodes(BiTree T)，返回二叉树T的非叶子结点数。

7．有下图所示的一个二叉树，请完成：

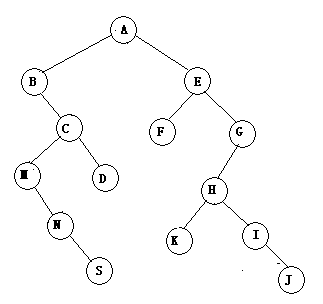
(1) 用顺序表表示法画出该二叉树的存储表示。

(2) 对该二叉树进行中序线索化，用二叉链表画出存储表示。



8．设哈希表HT表长m为13，哈希函数为H(k)=k MOD m，给定的关键值序列为{19,14,23,10,68,20,84,27,55,11}。试求出用线性探测再散列法解决冲突时所构造的哈希表，并求出在等概率的情况下查找成功的平均查找长度ASL。

1. 已知一组关键字序列为（12,5,9,20,6,31,24），对该组关键字序列进行升序排序，写出初始建好堆的关键字序列，依次写出堆排序每一趟排序的结果。
2. 将下图所示的二叉树还原成森林。



11.已知图C=(V，E)，其中V={a，b，c，d，e}，E={(a，b)，{a，c}，{a，d}，(b，c)，(d，c)，(b，e)，(c，e)，（d，e）}要求：

(1)画出图G；

(2)给出图G的邻接矩阵；

(3)给出图G的邻接表；

(4)给出图G的所有拓扑有序序列。

12．已知带权无向图的邻接表见下图，要求：

|  |  |
| --- | --- |
| a |  |
| b |  |
| c |  |
| d |  |
| e |  |
| f |  |
| g |  |

0

1

2

3

4

5

6

|  |  |  |
| --- | --- | --- |
| 1 | 9 |  |

|  |  |  |
| --- | --- | --- |
| 3 | 3 |  |

|  |  |  |
| --- | --- | --- |
| 4 | 2 |  |

|  |  |  |
| --- | --- | --- |
| 5 | 8 | ^ |

|  |  |  |
| --- | --- | --- |
| 0 | 9 |  |

|  |  |  |
| --- | --- | --- |
| 2 | 2 |  |

|  |  |  |
| --- | --- | --- |
| 5 | 3 | ^ |

|  |  |  |
| --- | --- | --- |
| 1 | 2 |  |

|  |  |  |
| --- | --- | --- |
| 3 | 7 |  |

|  |  |  |
| --- | --- | --- |
| 6 | 8 | ^ |

|  |  |  |
| --- | --- | --- |
| 2 | 7 |  |

|  |  |  |
| --- | --- | --- |
| 4 | 4 |  |

|  |  |  |
| --- | --- | --- |
| 6 | 9 | ^ |

|  |  |  |
| --- | --- | --- |
| 0 | 2 |  |

|  |  |  |
| --- | --- | --- |
| 3 | 4 |  |

|  |  |  |
| --- | --- | --- |
| 5 | 4 |  |

|  |  |  |
| --- | --- | --- |
| 6 | 1 | ^ |

|  |  |  |
| --- | --- | --- |
| 0 | 8 |  |

|  |  |  |
| --- | --- | --- |
| 1 | 3 |  |

|  |  |  |
| --- | --- | --- |
| 4 | 4 |  |

|  |  |  |
| --- | --- | --- |
| 6 | 5 | ^ |

|  |  |  |
| --- | --- | --- |
| 2 | 8 |  |

|  |  |  |
| --- | --- | --- |
| 3 | 9 |  |

|  |  |  |
| --- | --- | --- |
| 4 | 1 |  |

|  |  |  |
| --- | --- | --- |
| 5 | 5 | ^ |

|  |  |  |
| --- | --- | --- |
| 0 | 3 |  |

(1)画出图G。

(2)各画一棵从顶点a出发的深度优先生成树和广度优先生成树。

(3)给出用prim算法从顶点a出发构造最少生成树的过程。

(4)给出用Kruscal算法构造最小生成树的过程。

