2025数据结构回忆版

[注]打*号的题目数据未被记录，在保证**数据规模一致**的情况下由本人捏造

一、简答题(30分)

1.（5分）设有程序如下，分析其时间复杂度

```
void func(int n)
{
    int m = 0;
    for(int i = 1 ; i <= n ; i ++)
        for(int j = 1 ; j <= 2*i ; j ++) m++;
}
```

2.（5分）给定中缀表达式$a+b-a\times((c+d)/e-f)+g$

（1）写出与此中缀表达式等价的后缀表达式

（2）在中缀转后缀过程中，辅助栈的深度至少应该为多少

3.（5分）设p指针指向的双链表的中间节点（非头尾节点），写出在p节点**前**插入s指向的节点的流程和示意图，双链表节点定义如下

```
struct Node{
    int val;
    Node* prior;
    Node* next;
};
```

4.（5分）给定一颗二叉树的中序序列和后序序列

中序：$DFBEACHG$

后序：$FDEBHGCA$

（1）画出这颗二叉树

（2）写出这颗二叉树的前序序列

5.（5分）根据序列$\{46，78，35，48，70，99，26，54，66，121\}$构建一颗二叉搜索树并计算$ASL_{成功}$

6.（5分）无向图G的邻接矩阵（下三角）按照行优先存储在下列一维数组中（省略$a_{ii}=0$）

$$A=\{14,6,\infty,\infty,\infty,25,\infty,\infty,\infty,4,13,\infty,\infty,23,23\}$$

（1）画出该无向图G

（2）画出该无向图的DFS树

（3）画出该无向图的BFS树

二、应用题（40分）

1.（10分）设字符$\{a,b,c,d,e,f,g,h\}$出现的频率分别是$\{0.10,0.14,0.09,0.04,0.15,0.20,0.22,0.06\}$

（1）画出哈夫曼树（权值小的点放在左子树，编码左0右1）

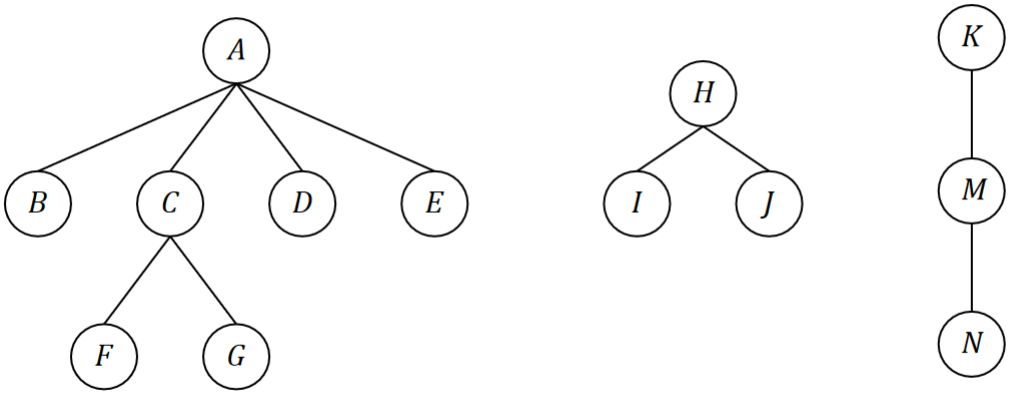
（2）计算该哈夫曼树的WPL，并计算相较于等长编码节省了多少比特

2.（10分）给定关键字序列$keys=\{36,26,44,28,45,33,68,12,23,66,25\}$，哈希函数为$H(key)=key \mod 11$

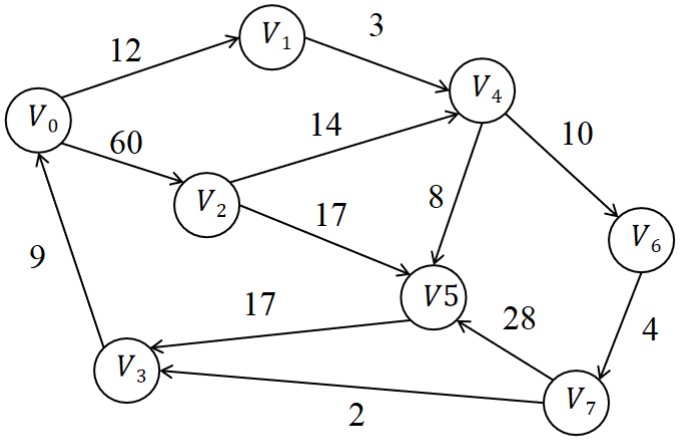
（1）画出哈希表（线性探测法解决冲突），并计算装填因子α

（2）计算等概率查找下的$ASL_{成功}$和$ASL_{失败}$

3.（10）将下列森林转化成二叉树



4*.（10）给定有向图G如下



（1）写出图G的邻接矩阵M

（2）用Dijkstra算法求源点V_0到其他点的最短路径，要求写出$T,dist,path$

三、算法设计题（30分）

1*.（15分）给定数组$A[n]$，对于$i<j$且$A[i]>A[j]$的点对$(A[i],A[j])$称为逆序对

（1）给出求解数组$A[n]$中逆序对数列的算法，求出序列$A[]=\{1,28,38,3,2,45,67,81,23,3,19,8\}$中的所有逆序对

（2）给出算法的首部如下，完善代码，要求时间复杂度$O(n\log n)$

```
int CountInversion(vector<int>& A,int Left,int Right)
```

2.（15分）给定二叉树的节点定义如下

```
struct TreeNode{
    char val;
    TreeNode* left;
    TreeNode* right;
    TreeNode(){left = right = NULL;}
    TreeNode(char c){val = c,left = right = NULL;}
};
```

（1）写出输出二叉树先序序列的代码，首部如下[可以在`leetcode144`提交]

```
void preorderTraversal(TreeNode* root)
```

（2）设二叉树中没有权值相同的节点，写出求值为c的节点的祖先个数，首部如下

```
int findAncestors(TreeNode* root,char c)
```

（3）写出求解二叉树宽度的算法，首部如下[可在`leetcode662`提交]

```
int widthOfBinaryTree(TreeNode* root)
```