## 名词解释

**算法：**对特定问题求解步骤的一种描述，是指令的有限序列。

**数据结构：**数据结构是**相互之间存在一种或多种特定关系的数据元素的集合**。包括3方面内容：逻辑结构、存储结构、对数据的运算。

**队列：**一种操作受限的线性表。其限制为允许在表的一端进行插入，而在另一端进行删除。其中允许插入的一段为队尾，进行删除的一端为队头。主要特点是：先进先出。

**二叉树：**二叉树是一种树形结构，其特点是每个结点至多有两棵子树，(即二叉树中的结点的度只能为0，1，2)其子树有左右之分，次序不能颠倒。

**串：**串是由零个或多个字符组成的有限序列，串中字符的个数称为串的长度，含有零个元素的串叫空串。

**栈：**栈是一种只能在一端进行插入或删除操作的线性表。其中栈顶允许插入或删除。主要特点是先进后出。

**有向图：**图中任意两个顶点之间的边都是有向边。（若是无向边则为无向图）。

**拓扑排序：**1）从有向图中选择一个没有前驱（入度为0）的顶点输出。2）删除1）中的顶点，并删除从该顶点出发的全部边。3）重复上述两步，直到剩余的网中不存在没有前驱的顶点为止。

**邻接矩阵：**图的顺序存储结构，由邻接矩阵的行数或列数可知图中的顶点数。

**堆排序：** 堆排序中的关键操作在于将序列调整为堆。整个排序过程就是通过不断调整使得不符合堆定义的完全二叉树变为符合堆定义的完全二叉树的过程。（堆：是一棵满足任何一个非叶结点的值都不大于或小于其左右孩子的值的完全二叉树）。

**关键路径：**从图中某一顶点（源点）出发到另一顶点（终点）的路径权值总和最大的路径。

**强连通图：**在有向图中，任意一对顶点vi到顶点vj和顶点vj到顶点vi都有路径。

**哈夫曼树：**在一棵二叉树中，若带权路径长度达到最小，称这样的二叉树为最优二叉树，即哈夫曼树。

**最小生成树：**n个顶点拥有n-1条边, 使得所有顶点之间都有路径可达，n个节点之间不能构成回路，且权值总和最小的生成树。

**拓展**

**图：**由顶点的有穷集合和顶点之间边的集合组成—表示为G（V，E）

**简单路径：**序列中顶点不重复出现的路径。

最短路径：从图中某一顶点（源点）出发到另一顶点（终点）的路径权值总和最小的路径。

## 简答题

1. ***什么是算法？算法的基本特征有哪些？***

算法是对特定问题求解步骤的一种描述，是指令的有限序列，其中每条指令表示一个或多个操作。算法具有以下5个特性

1. 有穷性：执行有限步后结束
2. 确定性：每一步骤必须有确定的含义
3. 可行性：算法中描述的操作都是可以通过已经实现的基本运算执行有限次来实现的
4. 输入：零个或多个输入，以刻画运算对象的初始情况
5. 输出：一个或多个输出，以反应对输入数据加工的结果。
6. ***线性表可用顺序表和链表存储，请问两种存储表示各有哪些优缺点？***
7. 基于空间的比较

1.1存储分配的方式：顺序表的存储空间是静态分配的。链表的存储空间是动态分配的

1.2存储密度（存储密度=结点值域所占的存储量/结点结构所占的存储量）：链表的存储密度=1，链表的存储密度<1（结点中有指针域）

1. 基于时间的比较

1.1存取方式：顺序表可以随机存取。链表是顺序存取的

1.2插入/删除时移动元素个数：顺序表需要移动近一半元素。链表不需要移动，只需要修改指针

1. ***什么是递归程序？递归程序的优缺点是什么？***

内部调用函数本身实现特定条件的程序称为递归程序。代码简洁，时间和空间消耗大，递归会有重复的计算。递归会存在栈溢出，每个进程的容量有限。

1. 简洁

2.在树的前序，中序，后序遍历[算法](http://lib.csdn.net/base/datastructure)中，递归的实现明显要比循环简单得多。

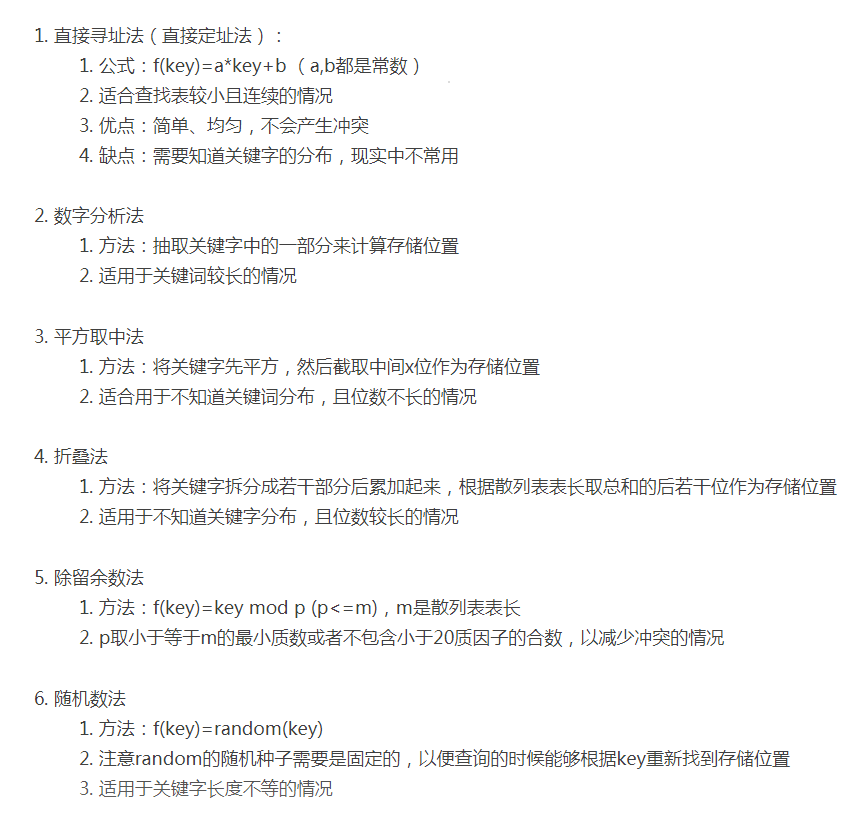
缺点：

1.递归由于是函数调用自身，而函数调用是有时间和空间的消耗的：每一次函数调用，都需要在内存栈中分配空间以保存参数、返回地址以及临时变量，而往栈中压入数据和弹出数据都需要时间。->效率

2.递归中很多计算都是重复的，由于其本质是把一个问题分解成两个或者多个小问题，多个小问题存在相互重叠的部分，则存在重复计算，如fibonacci斐波那契数列的递归实现。->效率

3.调用栈可能会溢出，其实每一次函数调用会在内存栈中分配空间，而每个进程的栈的容量是有限的，当调用的层次太多时，就会超出栈的容量，从而导致栈溢出。->性能

1. **哈希函数的构造方法有哪些？**



1. **简述什么是归并排序，其平均时间复杂度和最坏时间复杂度？**

归并排序：将两个或两个以上的有序子序列“归并”为一个有序序列

1. **数据元素之间的关系在计算机中有几种表示方法？各有什么特点？**

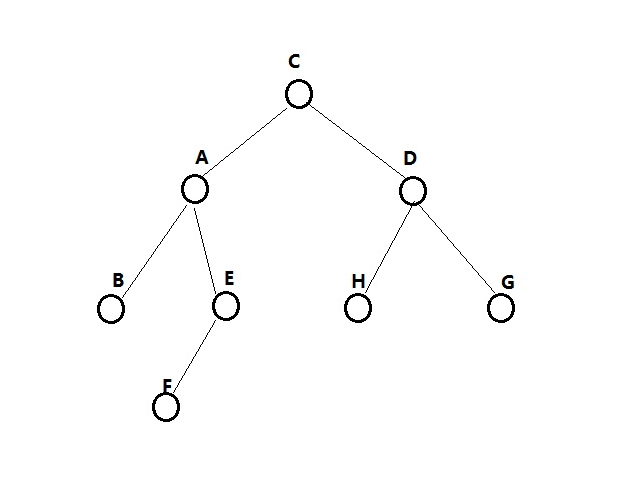
数据元素之间的关系在计算机中**有两种表示**方法：**顺序映像和非顺序映像**。对应的两种**存储结构**分别是***顺序存储结构*和*链式存储结构*。**顺序映像是借助**数据元素在存储器中的相对位置**来表示数据元素之间的逻辑关系。非顺序映像是指**借助指针**表示数据元素之间逻辑关系。

1. **结点数为n的各棵树中，深度最小的树的深度是多少？他有多少叶节点？多少分支节点？深度最大的树的深度是多少？它有多少叶节点？多少分支节点？**

高度最小的树的高度是1, 有2层, 有n-1个叶结点, 1个分支结点

高度最大的树的高度是n-1,有n层,有1个叶结点,n-1个分支结点

1. **如图是一颗二叉树，分别写出先序、中序、后序序列。**



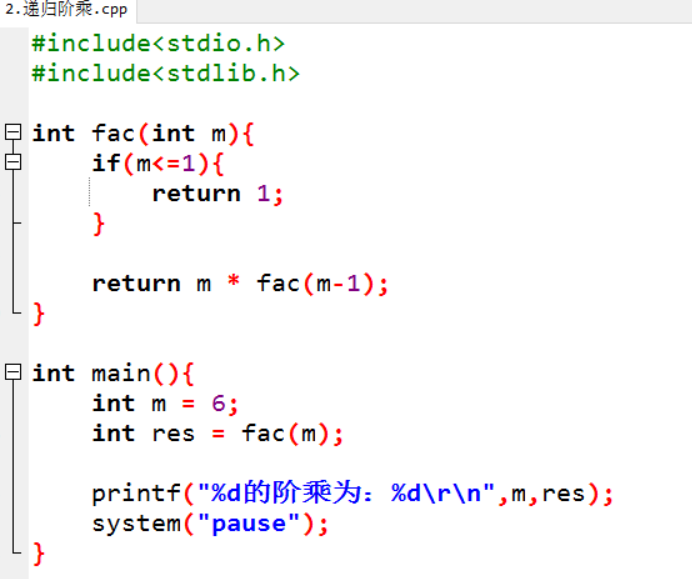
**先VLR: *CABEFDHG***

**中LVR: *BAFECHDG***

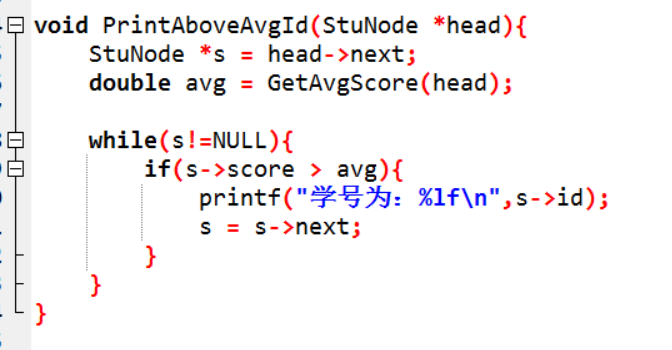
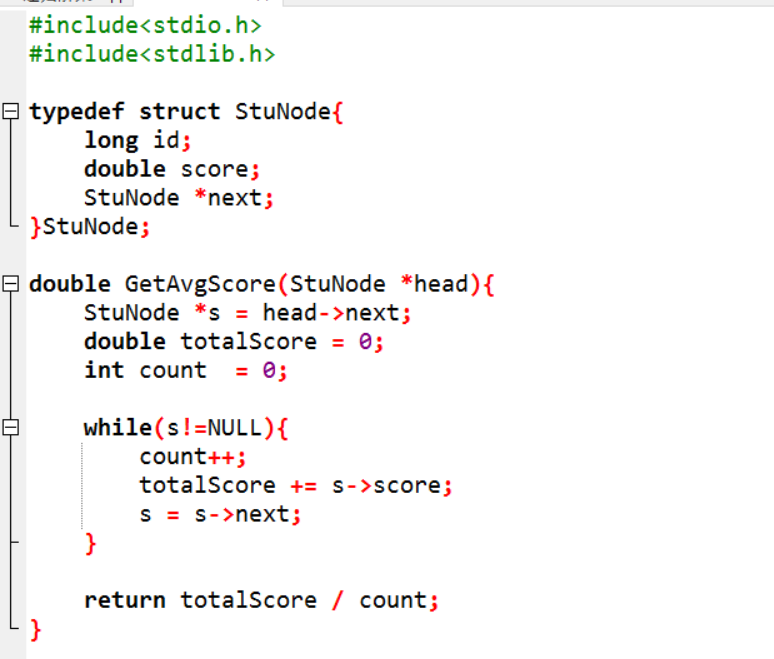
**后LRV: *BFEAHGDC***

## 程序题：

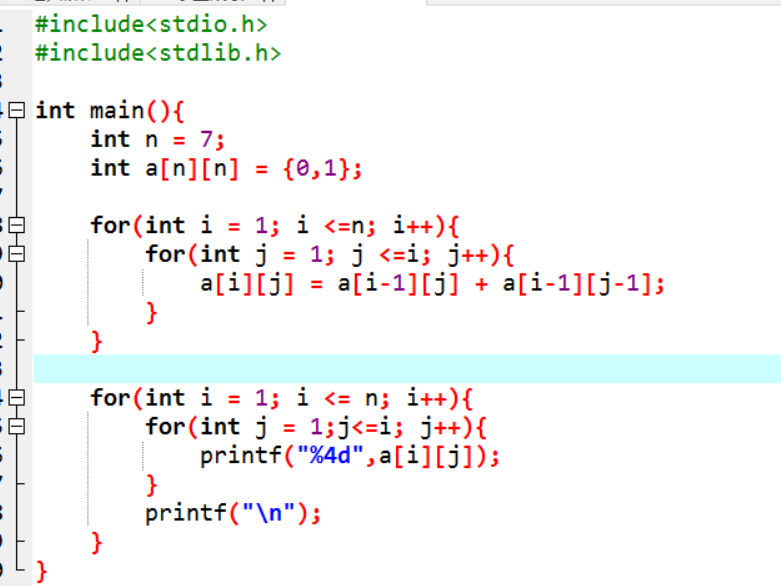
1. ***用递归实现m!的计算，并输出计算结果。（10分）***



1. ***有一个链表，设结点包含学号、成绩及指向下一结点的结构体指针：写一函数实现：输出成绩高于平均成绩的学生学号（要求：给出结构体定义）***



1. ***输出7行杨辉三角***



1. ***编写函数，统计出单链表HL中结点的值等于给定值x的结点数***

