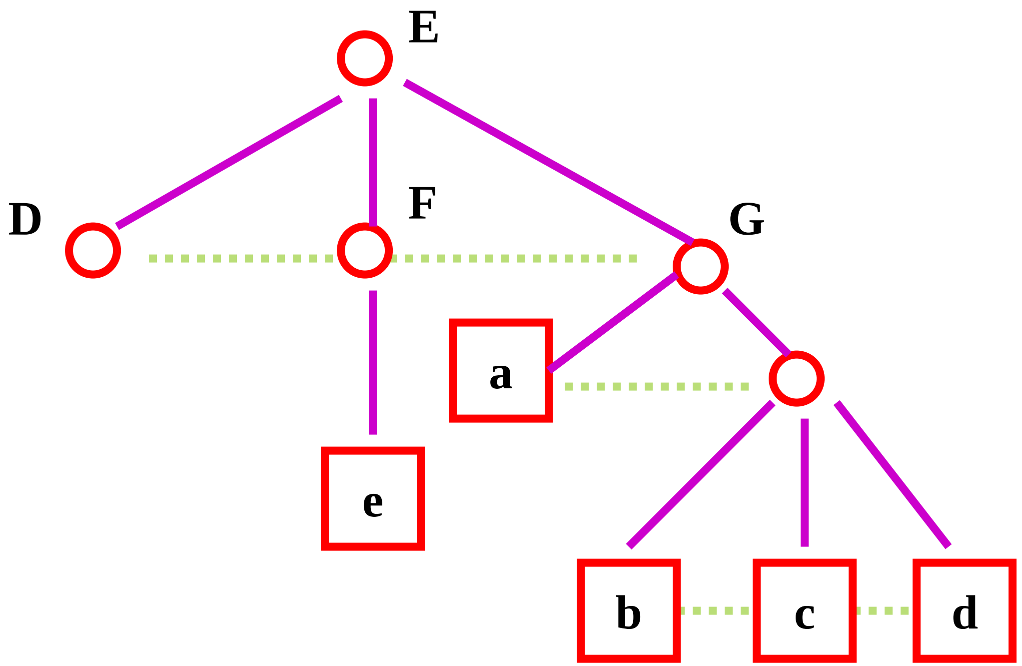
第6章 树和二叉树

第6章 树和二叉树

6.1 树的基本概念

E=(D,F,G)=(( ),(e),(a, (b,c,d)))

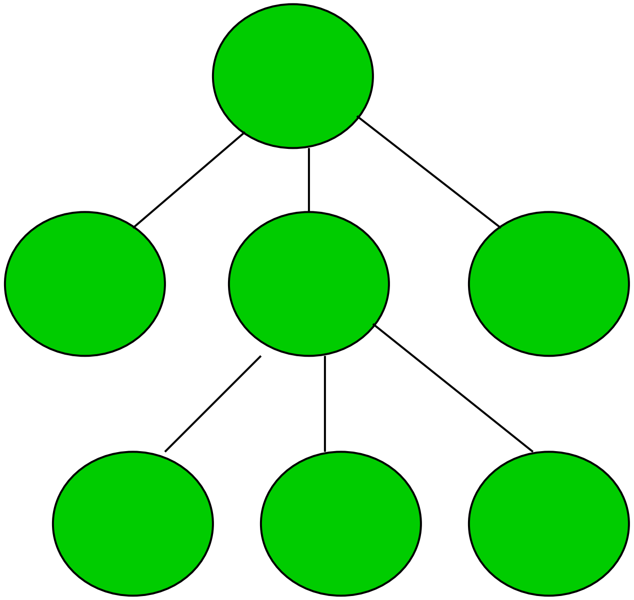


E的长度为3，深度为3

深度＝括号的重数＝ 结点的层数

6.1 树的基本概念

- 社会的组织机构
- 家族的族谱
- 计算机中的目录组织



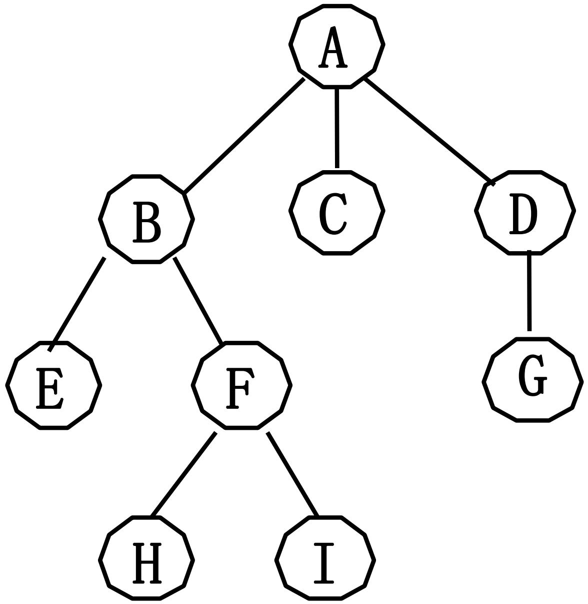
描述层次结构，是一种一对多的逻辑关系

6.1.1 树的定义

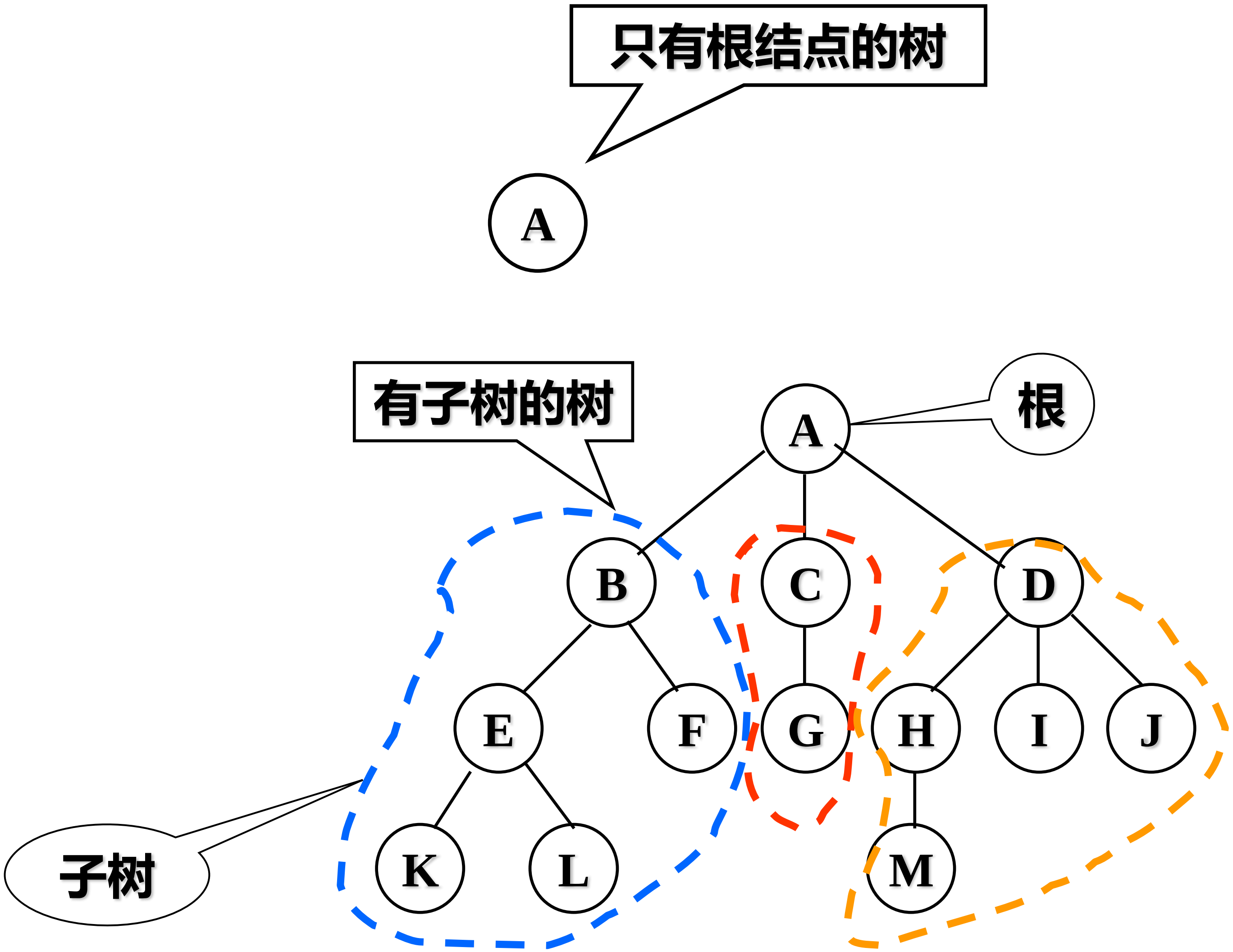
- 树(Tree)是n(n>=0)个结点构成的有限集合；
- n=0时称为空树；
- 非空树满足的条件：

  ① 有且仅有一个称为根（Root）的结点；

  ② n>1时，其余结点可分为m(m>0)个互不相交的有限集合$T_1 \ldots T_m$，其中每个集合又是一棵树，称为子树。



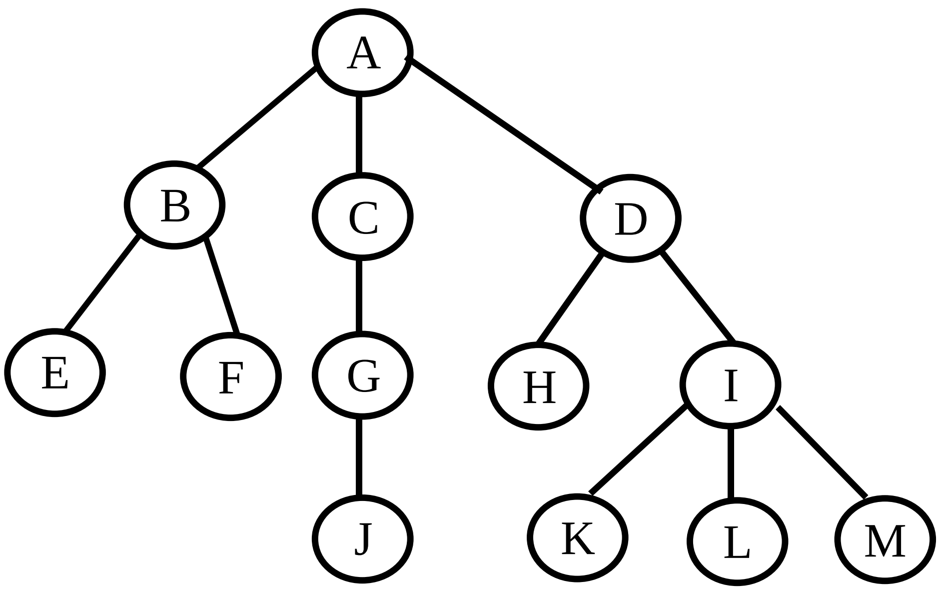
6.1.1 树的定义



6.1.2 树的逻辑表示方法

(1) 树形表示法

这是树的最基本的表示，使用一棵倒置的树表示树结构，非常直观和形象。



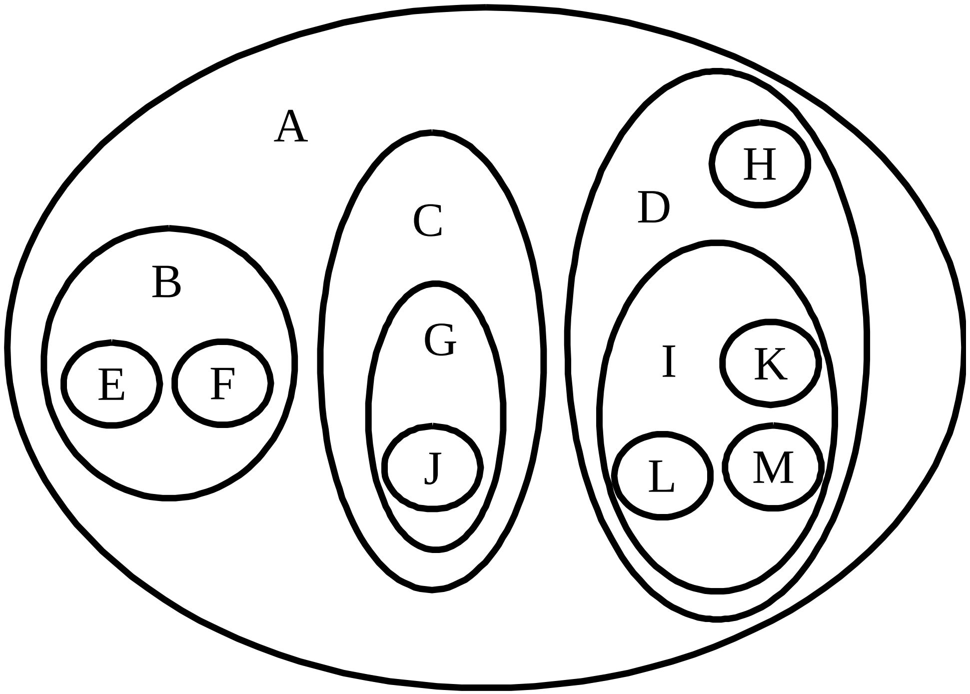
树形表示法

- 圆圈表示结点
- 圆圈内的符号代表该结点的数据信息
- 结点间的关系通过连线表示
- 连线隐含的方向为从上向下
- 上方结点为前驱，下方结点为后继

6.1.2 树的逻辑表示方法

(2)嵌套集合（文氏图表示法）

使用集合以及集合的包含关系描述树的结构。



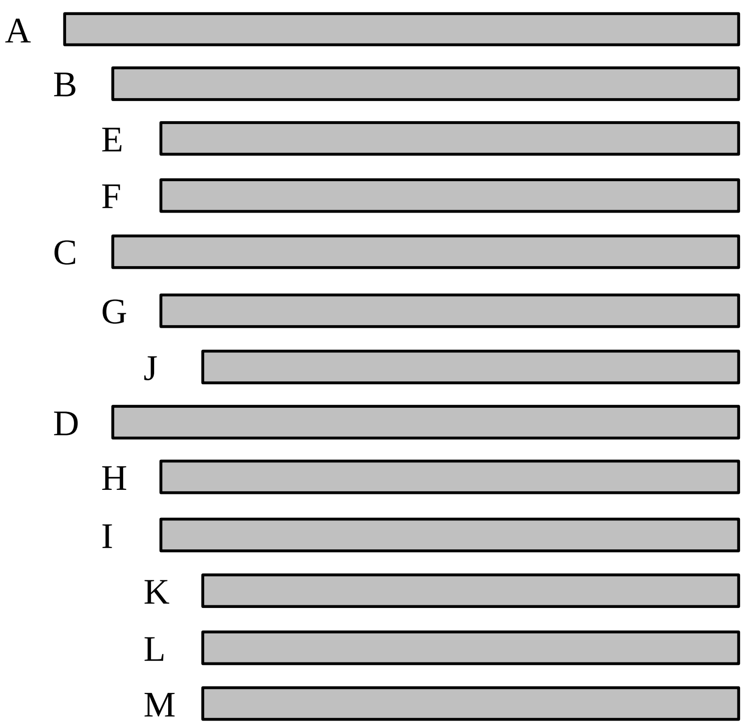
文氏图表示法

- 每棵树对应一个圆圈
- 圆圈内包含根结点和子树的圆圈
- 同一根结点下的各子树对应的圆圈不相交

6.1.2 树的逻辑表示方法

(3) 凹入表示法

使用线段的伸缩描述树的结构。



凹入表示法

- 每棵树的根对应着一个条形
- 子树的根对应着一个较短的条形
- 树根在上，子树的根在下
- 同一个根下的各子树的根对应的条形长度是一样的

6.1.2 树的逻辑表示方法

(4)广义表表示法

使用**括号的嵌套**描述树结构

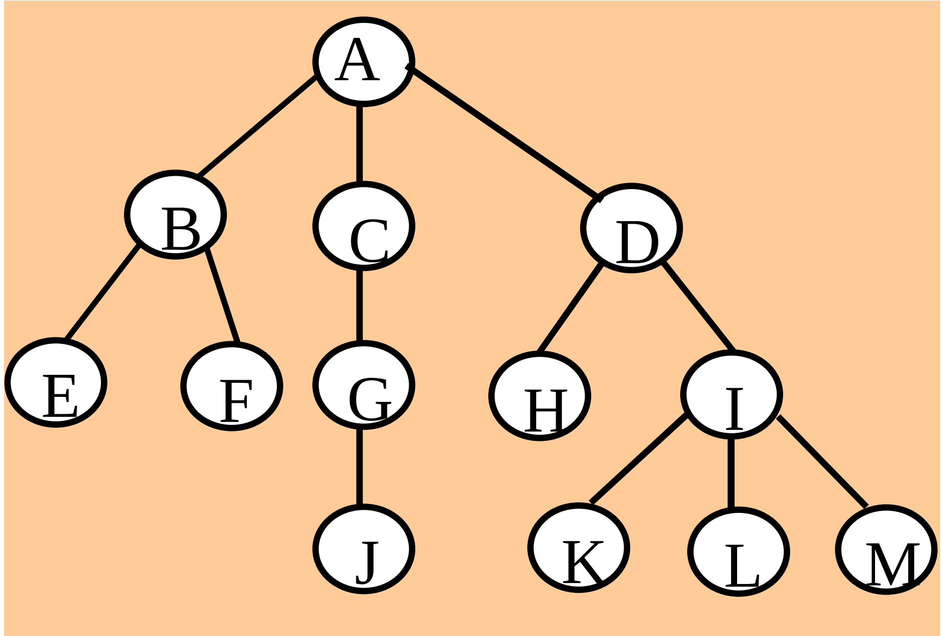
A(B(E,F),C(G(J)),D(H,I(K,L,M)))

括号表示法

- 每棵树对应一个表，根作为表的名字，写在括号的左边
- 除根结点之外的其余结点写在括号中，并用逗号间隔
- 括号里的各子树同样用表结构表示

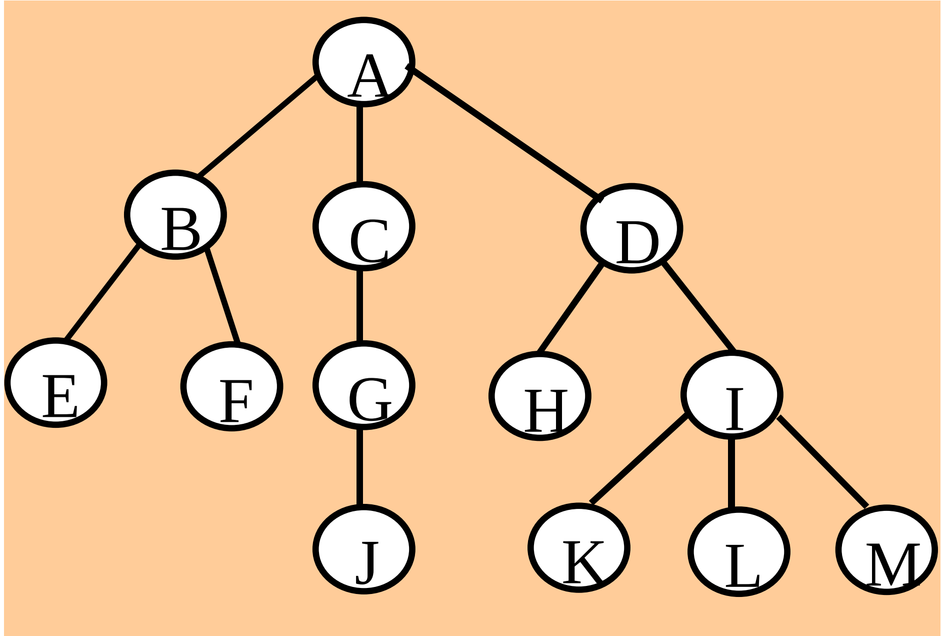
6.1.3 树的基本术语

- **结点的度：** 树中某结点的子树的个数。
- **树的度：** 树中各结点的度的**最大值**。
- **m次树：** 度为m的树。
- **分支结点：** 度不为零的结点称为非终端结点，又叫分支结点。
- **叶子结点：** 度为零的结点，又称为终端结点。
- 在分支结点中，每个结点的**分支数就是该结点的度。**



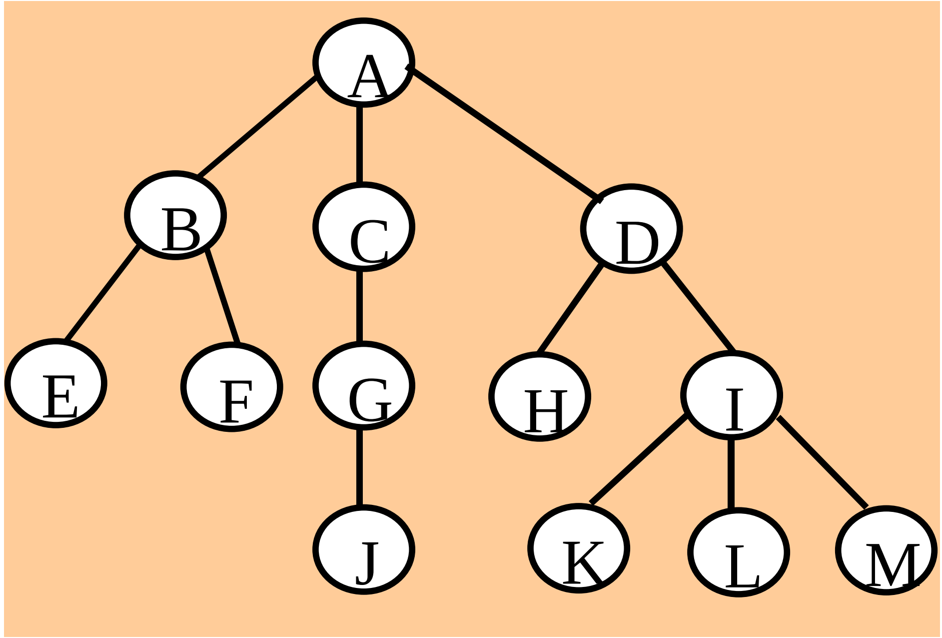
6.1.3 树的基本术语

- **路径**
  - 对于任意两个结点k_i和k_j，若树中存在一个结点序列$k_i,k_{i1},k_{i2},\ldots,k_{in},k_j$，使得序列中除$k_i$外的任一结点都是其在序列中的前一个结点的后继，则称该结点序列为由k_i到k_j的一条路径。
  - 用路径所通过的**结点序列**$(k_i,k_{i1},k_{i2},\ldots,k_j)$表示这条路径。
  - 路径就是从k_i出发"自上而下"到达k_j所通过的树中结点序列
- **路径长度**
  - 从根结点到某结点的边数
  - 路径通过的**结点数目减1**



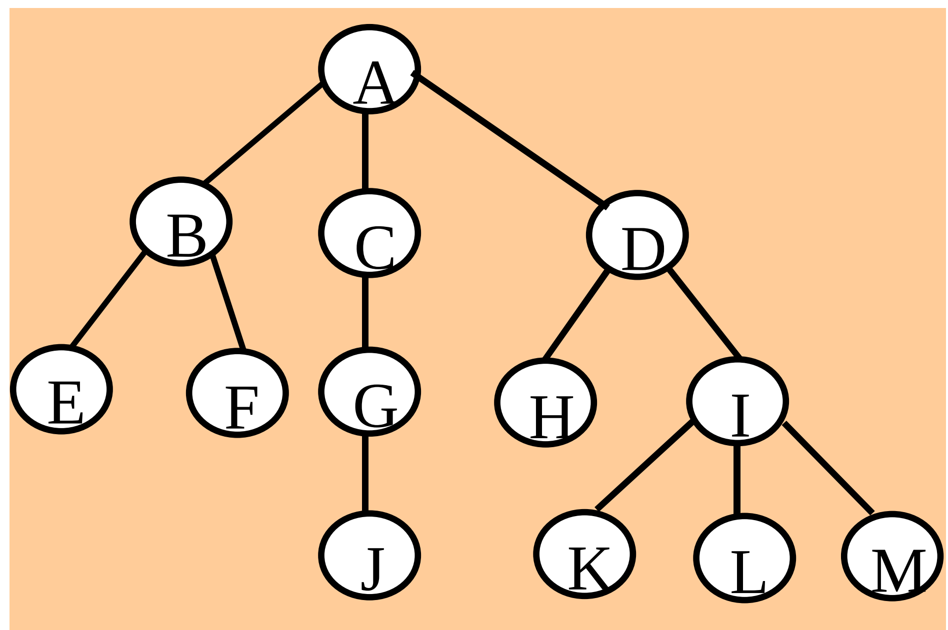
6.1.3 树的基本术语

- 孩子，双亲，兄弟
  - 在一棵树中，每个结点的后继，被称作该结点的孩子结点(或子女结点)。
  - 该结点被称作孩子结点的双亲结点(或父母结点)。
  - 具有同一双亲的孩子结点互为兄弟结点。
- 子孙结点，祖先结点
  - 每个结点的所有子树中的结点称为该结点的子孙结点。
  - 从树根结点到达该结点的路径上经过的所有结点被称作该结点的祖先结点。



6.1.3 树的基本术语

- **结点的层次**
  - 从树根开始定义，根结点为第1层，它的孩子结点为第2层，以此类推，一个结点所在的层次为其双亲结点所在的层次加1。
- **树的深度：** 树中结点的**最大**层次，也称为树的高度。
- **有序树和无序树**
  - 若树中各结点的子树是按照一定的次序从左向右安排的，且相对次序**不能随意变换**，则称为有序树，否则称为无序树。
- **森林：** m(m>=0)个互不相交的树的集合。



6.1.3 树的基本术语

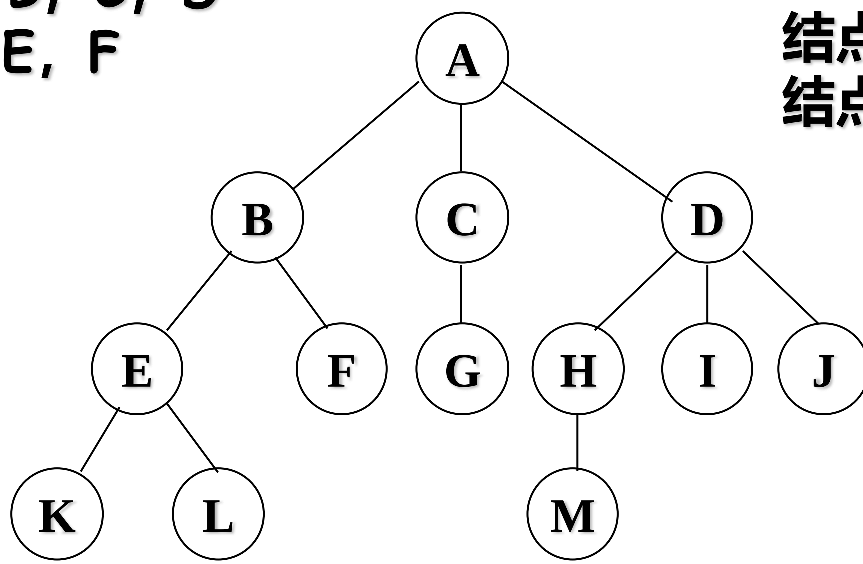
结点A的度：3
结点B的度：2
结点M的度：0

叶子：K，L，F，G，M，I，J

结点I的双亲：D
结点L的双亲：E

结点A的孩子：B，C，D
结点B的孩子：E，F

结点B，C，D为兄弟
结点K，L为兄弟



树的度：3

树的深度：4

结点A的层次：1
结点M的层次：4

结点F，G为堂兄弟
结点A是结点F，G的祖先

6.1.3 树的基本术语

线性结构	树形结构
唯一的第一个数据元素 (无前驱)	根结点 (无前驱)
唯一的最后一个数据元素 (无后继)	多个叶子结点 (无后继)
其它数据元素 (一个前驱 一个后继)	其它内部结点 (一个前驱 多个后继)

6.1.4 树的抽象数据类型

ADT Tree{

数据对象：$D=\{a_i|1\le i\le n$，$n\ge 0$，a_i属ElemType类型$\}$

数据关系：$R=\{<a_i,a_j>|a_i,a_j\in D,1\le i\le n, 1\le j\le n$,其中每个元素只有一个直接前驱，可以有零个或多个直接后继，有且仅有一个元素没有直接前驱$\}$

基本运算：

```
InitTree(&t);          //初始化树：构造一棵空树t
DestroyTree (&t);      //销毁树：释放树t占用的存储空间
Parent(t);             //求t所指结点的双亲结点（直接前驱）
Sons(t);               //求t所指结点的子孙结点（所有后继）
…
```

}ADT Tree

6.1.5 树的基本运算

树的运算主要分为四大类：

第一类，常规操作，树的初始化和销毁等；

第二类，寻找满足某种特定关系的结点，如寻找当前结点的双亲结点等；

第三类，插入或删除某个结点，如在树的当前结点上插入一个新结点或删除当前结点的第i个孩子结点等；

第四类，遍历树中每个结点。

6.2 二叉树

6.2.1 二叉树的概念

6.2.2 二叉树的性质

6.2.3 二叉树与树、森林之间的转换

6.2.1 二叉树的概念

甲：猜个100以内的数字？

乙：50

甲：小了

乙：75

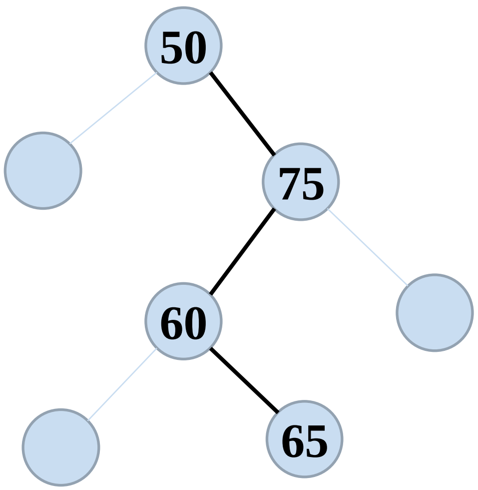
甲：大了

乙：60

甲：小了

乙：65

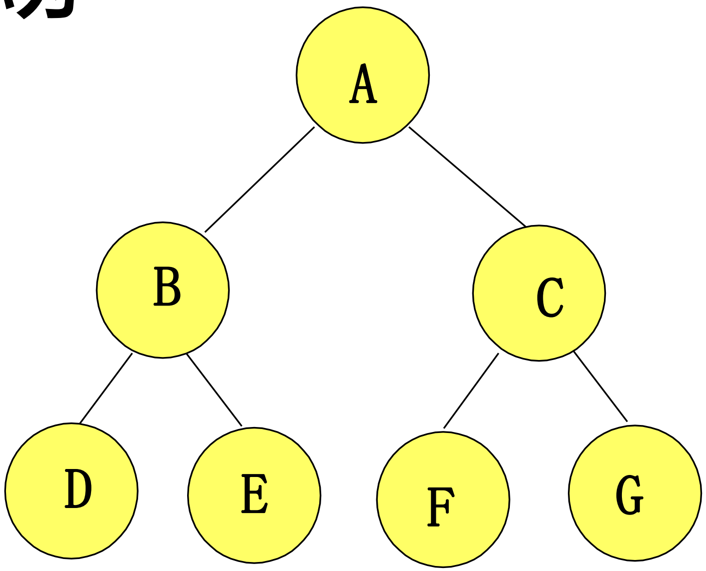
甲：正确



6.2.1 二叉树的概念

- **定义：**

一棵二叉树是结点的一个有限集合，该集合或者为空，或者是由一个根结点加上两棵分别称为左子树和右子树的、互不相交的二叉树组成。



- **特征：**

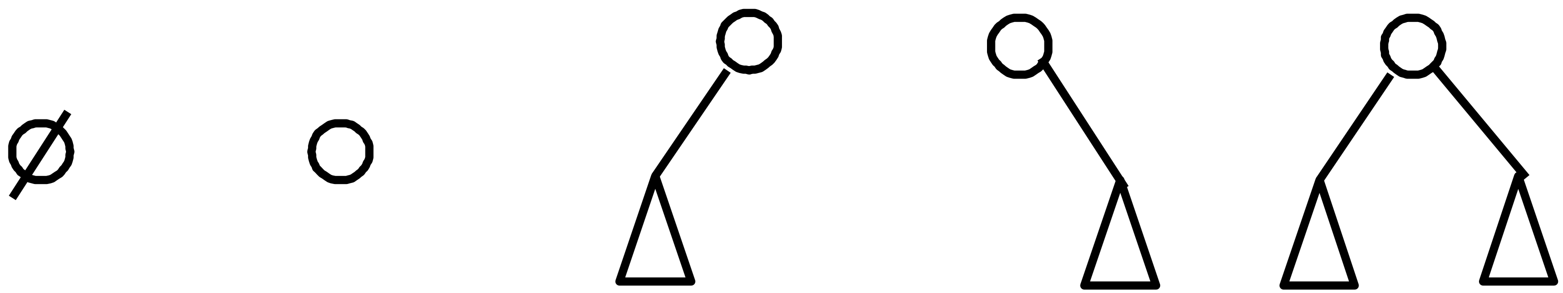
1）度≤2：

每个结点最多只有两棵子树

2）有序树：**子树有左右之分，其次序不能任意颠倒**

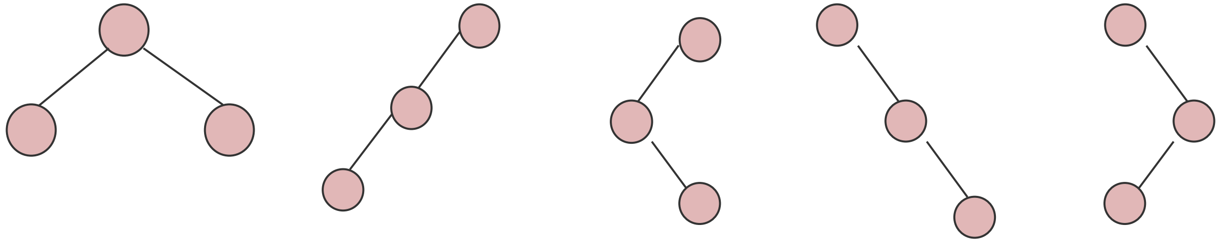
6.2.1 二叉树的概念

二叉树的五种形态



(a)　(b)　(c)　(d)　(e)

问：具有3个结点的二叉树可能有几种不同形态?



有5种

6.2.1 二叉树的概念

二叉树和树区别

- 区别
  - ✓ 二叉树可以为空，树不能
  - ✓ 二叉树每个节点的度不大于 2，树则没有限制
  - ✓ 二叉树是有序的，树则分为有序和无序两种
- 二叉树和度为2的树的区别
  - ✓ 度为2的树中至少有一个结点的度为2，二叉树没有此要求
  - ✓ 度为2的树不区分左右子树，二叉树严格区分
- 二叉树 与 度最大为2的有序树

6.2.1 二叉树的概念

ADT BinaryTree{

数据对象D: D是具有相同特性的数据元素的集合。

数据关系R: 若D=Φ，则R= Φ；

若D≠Φ，则R= {H}；存在二元关系：

① root 唯一 //关于根的说明

② $D_j \cap D_k = \Phi$ //关于子树不相交的说明

③ …… //关于数据元素的说明

④ …… //关于左子树和右子树的说明

基本操作 P: //至少有20个，如返回某结点的左孩子，或中序遍历，等等

}ADT BinaryTree

6.2.2 二叉树的性质

性质1：在二叉树的第 i 层上至多有 2^{i-1} 个结点 ($i \geq 1$)。

用归纳法证明:

归纳基： $i = 1$ 层时，只有一个根结点：

$$2^{i-1} = 2^0 = 1;$$

归纳假设： 假设对所有的 j，$1 \leq j < i$，命题成立；

归纳证明： 当 $j = i-1$ 时，命题成立，最多有 2^{i-2} ***个节点***

二叉树上每个结点至多有两棵子树，

则第 i 层的结点数 $= 2^{i-2} \times 2 = 2^{i-1}$ 。

6.2.2 二叉树的性质

性质2：深度为 *k* 的二叉树上至多含 2^k-1 个结点（k≥1）。

证明:

基于上一条性质，深度为 *k* 的二叉树上的结点数至多为

$2^0+2^1+\cdots\cdots+2^{k-1} = 2^k-1$ 。

(等比数列求和)

$$\sum_{i=1}^{k}(\text{第}i\text{层的最大结点数}) = \sum_{i=1}^{k}2^{i-1} = 2^k - 1$$

6.2.2 二叉树的性质

性质3：对任何一棵二叉树，若它含有n_0个叶子结点（0度节点）、n_2 个度为 2 的结点，则必存在关系式： $n_0 = n_2+1$ 。

证明：设 二叉树上结点总数 $n = n_0 + n_1 + n_2$

又 二叉树上分支总数 $b = n_1+2n_2$

而 $b = n-1 = n_0 + n_1 + n_2 - 1$

由此， $n_0 = n_2 + 1$ 。

同理： 三次树： $n_0=1+n_2+2n_3$

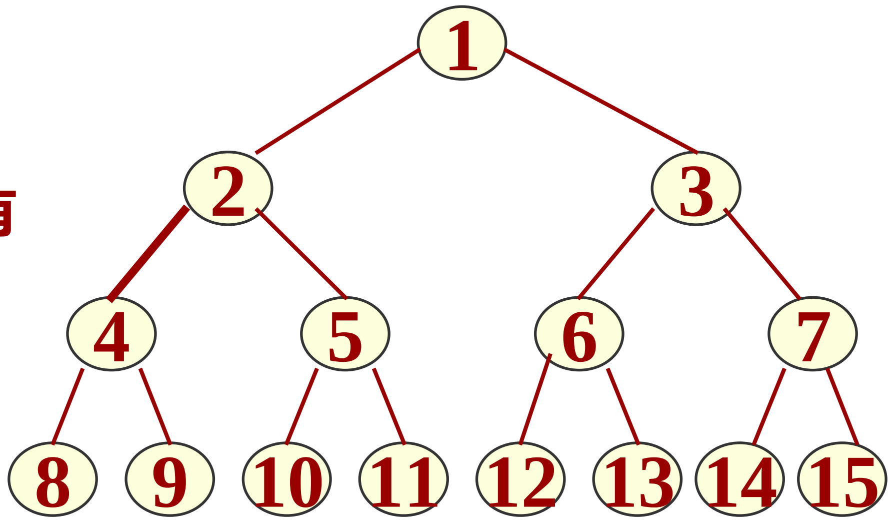
四次树： $n_0=1+n_2+2n_3+3n_4$

…

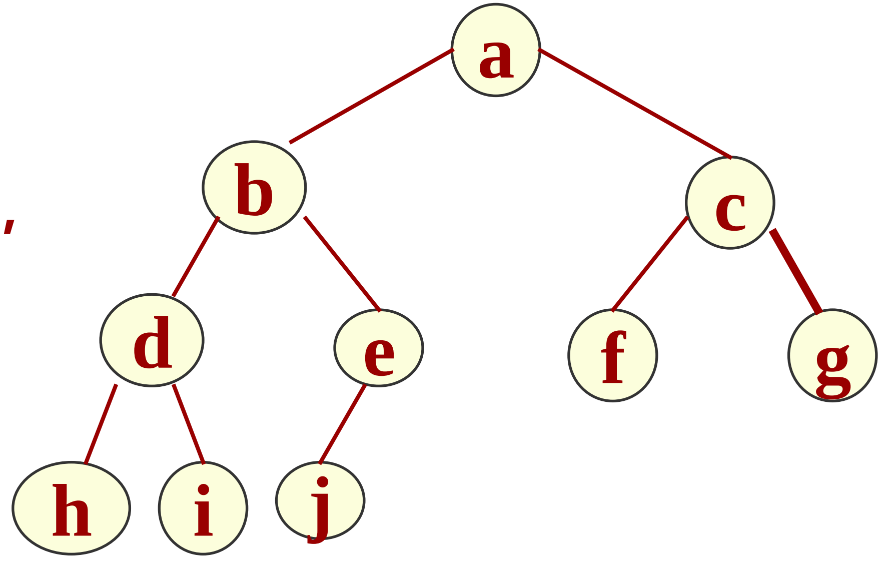
K次树： $n_0=1+n_2+2n_3+\cdots+(k-1)n_k$

6.2.2 二叉树的性质

两类特殊的二叉树：



满二叉树：指的是深度为*k*且含有2^k-1个结点的二叉树。

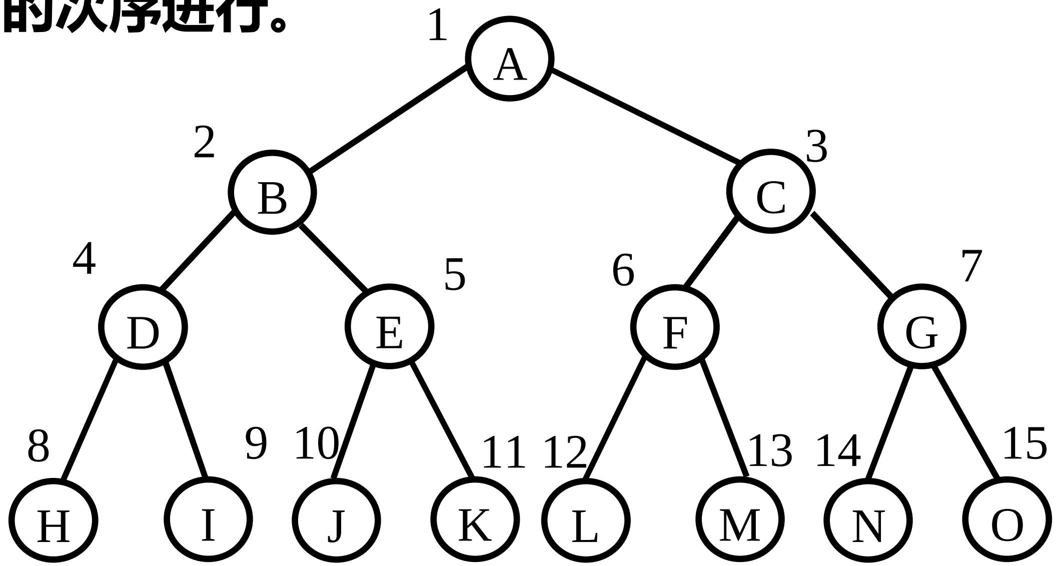


完全二叉树：树中所含的 *n* 个结点和满二叉树中编号为 *1* 至 *n* 的结点一一对应。（编号的规则为，由上到下，从左到右。）

6.2.2 二叉树的性质

满二叉树

- **所有分支结点都有左孩子结点和右孩子结点，并且叶结点都集中在二叉树的最下一层；**
- **深度为k且有2^k-1个结点；**
- **每一层上的结点数都是最大结点数；**
- **不存在度为1的结点。**
- **可以对满二叉树的结点进行连续编号，约定编号从树根为1开始，按照层数从小到大、同一层从左到右的次序进行。**



6.2.2 二叉树的性质

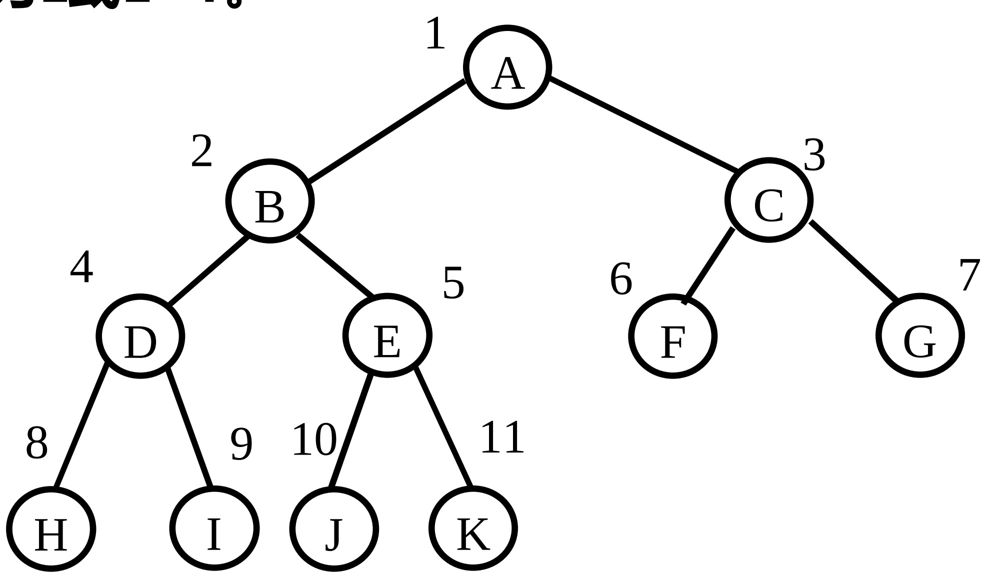
完全二叉树

定义：深度为k，有n个结点的二叉树，当且仅当其每一个结点的位置序号都与深度为k的满二叉树的结点编号一一对应时，成为完全二叉树。

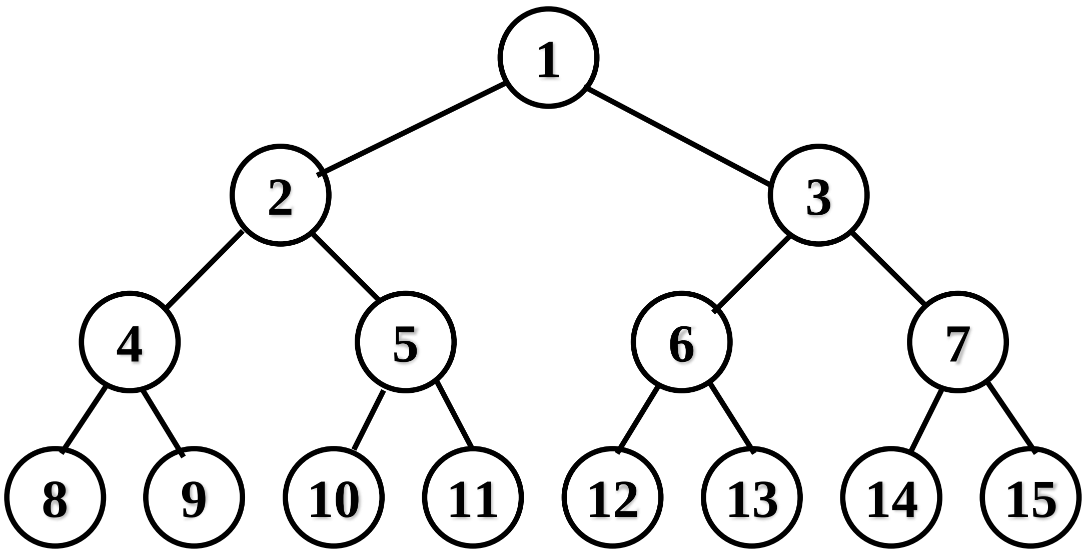
特点：

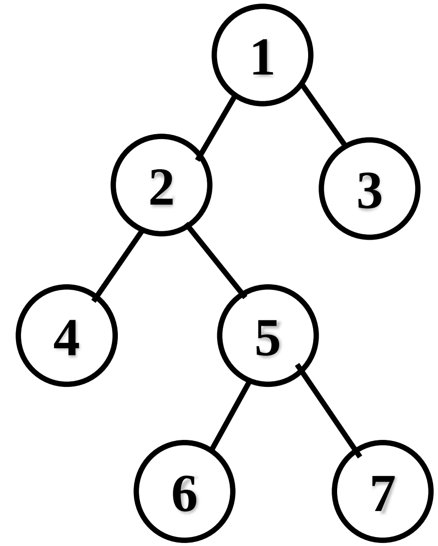
（1）叶子结点只可能在层次最大的两层上出现。

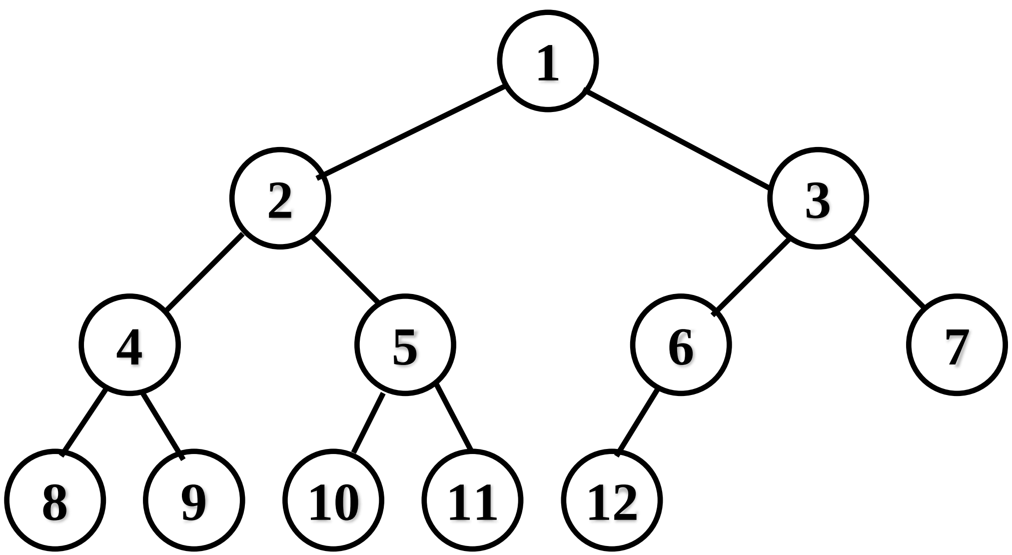
（2）对任一结点，若其右分支下子孙的最大层次为L，则其左分支下子孙的最大层次必为L或L+1。

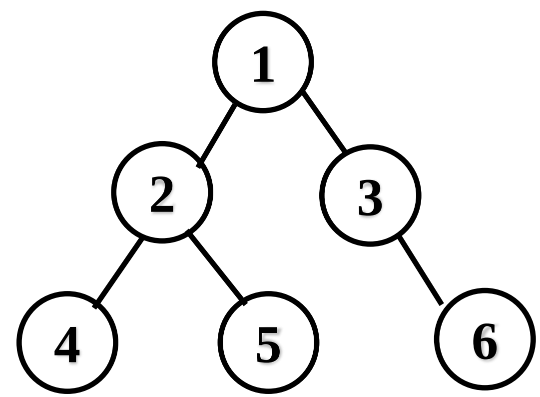


6.2.2 二叉树的性质









6.2.2 二叉树的性质

性质4　具有n个(n > 0)结点的完全二叉树的s深度为$\lfloor log_2 n \rfloor+1$。

证明:

设完全二叉树的深度为 k

则根据第二条性质得$2^{k-1}-1 < n \leqslant 2^k -1$

或　$2^{k-1} \leqslant n < 2^k$

即　$k-1 \leq log_2 n < k$

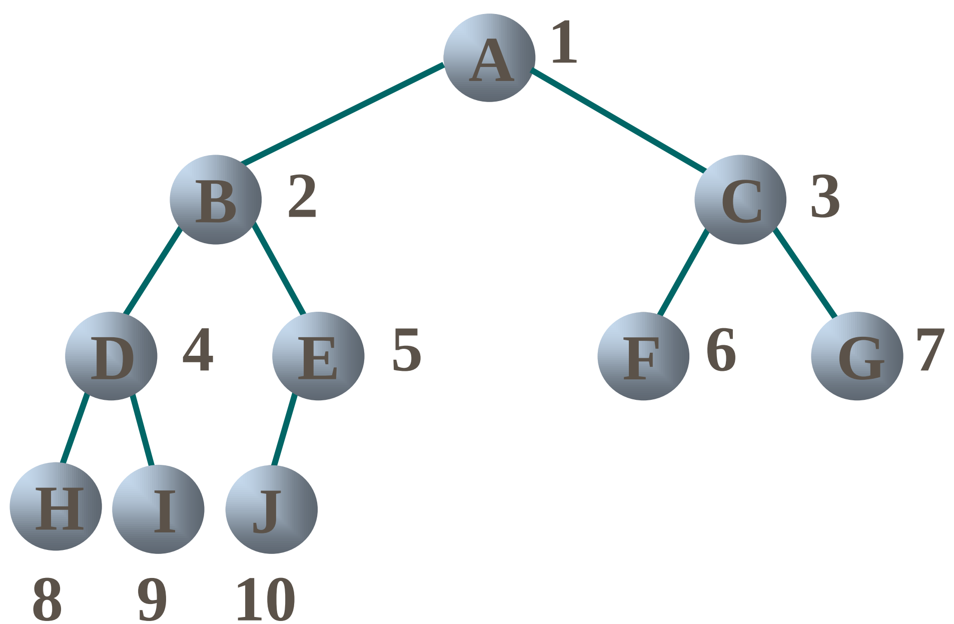
因为 k 只能是整数，因此， $k = \lfloor log_2 n \rfloor + 1$

$k = \lceil log_2(n+1) \rceil$

6.2.2 二叉树的性质

性质5 若对含 *n* 个结点的完全二叉树从上到下且从左至右进行 *1* 至 *n* 的编号，则对完全二叉树中任意一个编号为 *i* 的结点：

(1) 若 $i=1$，则该结点是二叉树的根，无双亲，否则，编号为 $\lfloor i/2 \rfloor$ 的结点为其双亲结点；

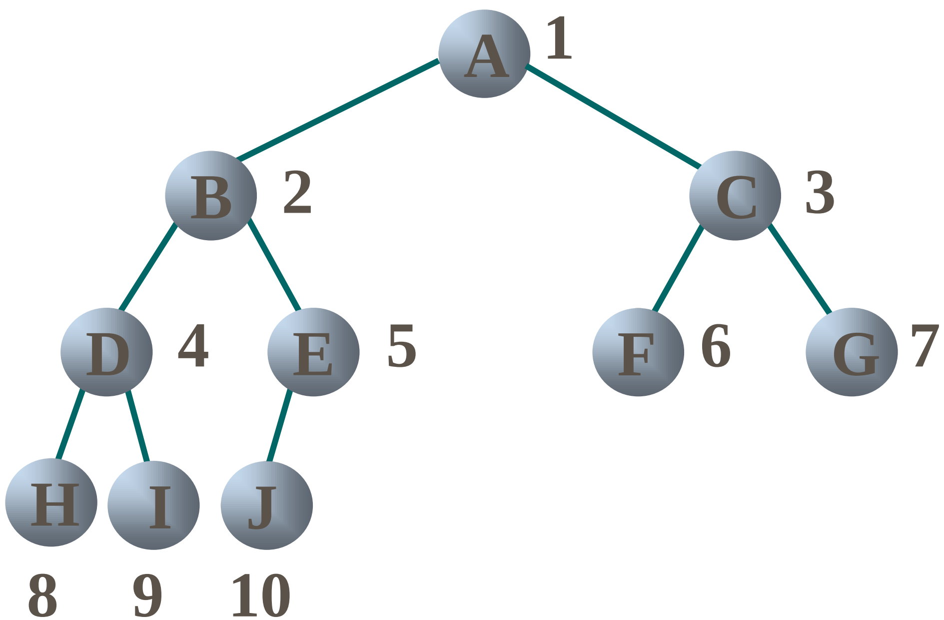


6.2.2 二叉树的性质

性质5 若对含 *n* 个结点的完全二叉树从上到下且从左至右进行 *1* 至 *n* 的编号，则对完全二叉树中任意一个编号为 *i* 的结点：

(2) 若 $2i>n$，则该结点无左孩子，

否则，编号为 $2i$ 的结点为其左孩子结点；

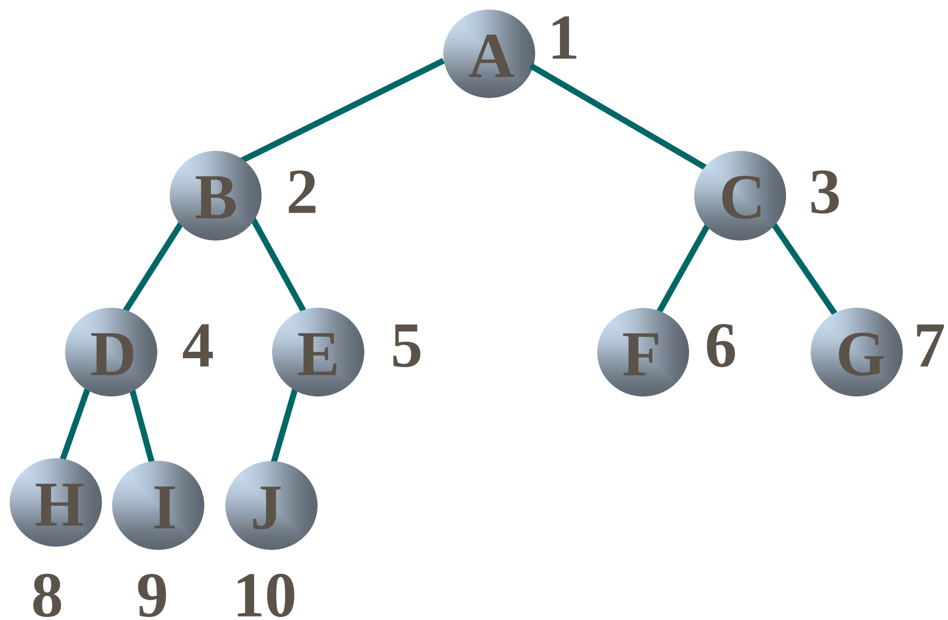


6.2.2 二叉树的性质

性质5 若对含 *n* 个结点的完全二叉树从上到下且从左至右进行 *1* 至 *n* 的编号，则对完全二叉树中任意一个编号为 *i* 的结点：

(3) 若 $2i+1>n$，则该结点无右孩子结点，

否则，编号为$2i+1$ 的结点为其右孩子结点。



6.2.2 二叉树的性质

性质5 若对含 *n* 个结点的完全二叉树从上到下且从左至右进行 *1* 至 *n* 的编号，则对完全二叉树中任意一个编号为 *i* 的结点：

(1) 若 $i=1$，则该结点是二叉树的根，无双亲，
否则，编号为 $\lfloor i/2 \rfloor$ 的结点为其**双亲**结点；

(2) 若 $2i>n$，则该结点无左孩子，
否则，编号为 $2i$ 的结点为其**左孩子**结点；

(3) 若 $2i+1>n$，则该结点无右孩子结点，
否则，编号为 $2i+1$ 的结点为其**右孩子**结点。

- 若 i 为偶数，且i != n，则其右兄弟为i+1
  若 i 为奇数，且i != 1，则其左兄弟为i-1
- i 所在层次为 $\lfloor \log_2 i \rfloor + 1$

6.2.2 二叉树的性质

完全二叉树性质推论

- n个结点的完全二叉树中：
  - ✓ 度为1的结点数为 $(n+1)\%2$ ——n为奇数时为0，n为偶数时为1
  - ✓ 度为0的结点数为 $\lfloor (n+1)/2 \rfloor$
  - ✓ 度为2的结点数为 $\lfloor (n+1)/2 \rfloor -1$
  - ✓ 编号最大的分支结点是 $\lfloor n/2 \rfloor$
  - ✓ 编号最小的叶子结点是 $\lfloor n/2 \rfloor +1$
- 具有n_0个叶子结点的完全二叉树中共有$2n_0$个结点或$2n_0-1$个结点。

6.2.2 二叉树的性质

例：请计算完全二叉树双亲节点、孩子节点及所在层次

i=7，n=12

双亲节点：$\lfloor i/2 \rfloor=3$

2i>成立：无孩子节点

所在层次：$\lfloor \log_2 i \rfloor+1=3$

i=5，n=12

双亲节点：$\lfloor i/2 \rfloor=2$

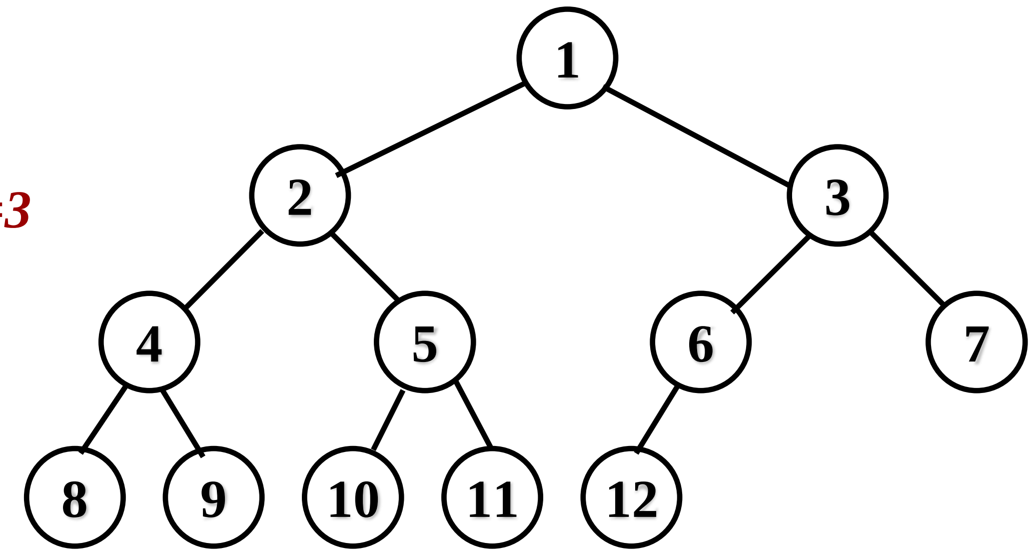
2i>n不成立：

左孩子：2i=10

2i+1>n不成立：

右孩子：2i+1=11

所在层次：$\lfloor \log_2 i \rfloor+1=3$



6.2.2 二叉树的性质

例：在下述结论中，正确的是(　　)

①只有一个结点的二叉树的度为0;

②二叉树的度为2;

③二叉树的左右子树可任意交换;

④深度为K的完全二叉树的结点个数小于或等于深度相同的满二叉树。

6.2.2 二叉树的性质

一棵完全二叉树有1000个结点，则它必有______个叶子结点，有______个度为2的结点，有______个结点只有非空左子树，有______个结点只有非空右子树。

分析题意：已知n=1000,求n_0和n_2,还要判断末叶子是挂在左边还是右边?

请注意：叶子结点总数≠末层叶子数！！！

深度$\lfloor log_2 n \rfloor+1=10$

前9层$2^k-1=511$

第10层叶子节点1000-511=489

第9层叶子节点256-（244+1）=11

$n_0 = n_2+1$

正确答案：

全部叶子数=489+11=500个。

度为2的结点＝叶子总数－1=499个。

最后一结点为2i属于左叶子，右叶子是空的，所以有1个非空左子树。完全二叉树的特点决定不可能有左空右不空的情况，所以非空右子树数＝0。

6.2.2 二叉树的性质

一颗二叉树第六层（即深度为6）的节点树最多为?

二叉树每层的节点数最多为$2^{(k-1)}$;

第六层节点最多为：$2^5=32$

某二叉树中度为2的节点有18个，则该二叉树中有多少个叶子节点?

$n_0 = n_2+1$;

叶子节点数为：18+1=19

具有53个节点的完全二叉树的深度为?

$2^{(k-1)} - 1 < 53$;

取最大的k值：6

6.2.2 二叉树的性质

在具有 2n 个结点的完全二叉树中，叶子结点个数为?

$2n = n_0 + 1 + n_2 = n_0 + 1 + (n_0 - 1)$;

叶子结点： n

在一颗度为3的树中，度为3的结点有2个，度为2的结点有1个，度为1的结点有2个，则叶子结点有（ ）个

$n_0=1+n_2+2n_3$;

叶子节点数为：n_0 =1+1+2*2=6

正在答疑