树的定义

树是n(n>=0)个结点的有限集。当n=0时，称为空树。在任意一棵非空树中应满足：

1. 有且仅有一个特定的称为根的结点。
2. 当n>1时，其余结点可分为m(m>0)个互不相交的有限集T1,T2,...,Tm，其中每个集合本身又是一棵树，并且称为根的子树。

显然，树的定义是递归的，即在树的定义中又用到了其自身，树是一种递归的数据结构。树作为一种逻辑结构，同时也是一种分层结构，具有以下两个特点：

1. 树的根结点没有前驱，除根结点外的所有结点有且只有一个前驱。
2. 树中所有结点都可以有零个或多个后继。

树适用于表示具有层次结构的数据。树中的某个结点（除根结点外）最多只和上一层的一个结点（即其父结点）有直接关系，根结点没有直接上层结点，因此在n个结点的树中有n-1条边。而树中每个结点与其下一层的零个或多个结点（即其孩子结点）都有直接关系。

基本术语

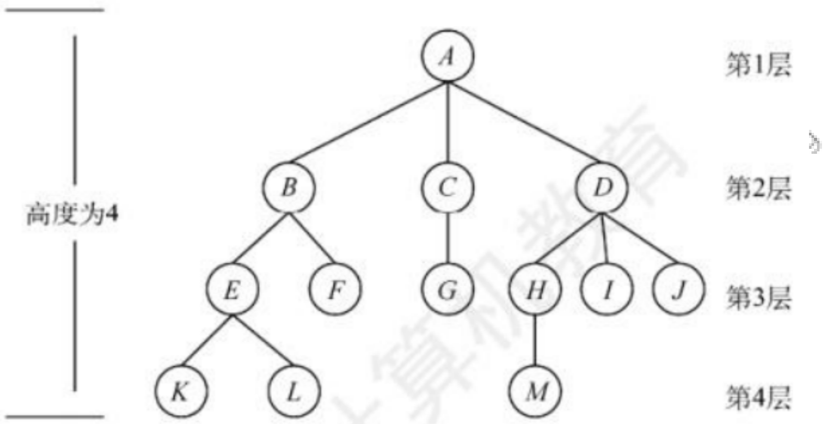

图 5.1 树的树形表示

祖先子孙双亲孩子兄弟堂兄弟	考虑结点K，从根A到结点K的唯一路径上的所有其他结点，称为结点K的祖先。如结点B是结点K的祖先，而K是B的子孙，结点B的子孙包括E,F,K,L。路径上最接近结点K的结点E称为K的双亲，而K为E的孩子。根A是树中唯一没有双亲的结点。有相同双亲的结点称为兄弟，如结点K和结点L有相同的双亲E，即K和L为兄弟。双亲在同一层的结点互为堂兄弟，结点G与E,F,H,I,J互为堂兄弟。

结点的度树的度	树中一个结点的孩子个数称为该结点的度，树中结点的最大度数称为树的度。如结点B的度为2，结点D的度为3，树的度为3。
分支结点叶结点	度大于0的结点称为分支结点（又称非终端结点）；度为0（没有孩子结点）的结点称为叶结点（又称终端结点）。在分支结点中，每个结点的分支数就是该结点的度。
结点的深度结点的高度结点的层次	结点的层次从树根开始定义，根结点为第1层，它的孩子为第2层，以此类推。结点的深度就是结点所在的层次。树的高度（或深度）是树中结点的最大层数。结点的高度是以该结点为根的子树的高度。图中树的高度为4。
有序数无序树	树中结点的各子树从左到右是有次序的，不能互换，称该树为有序树，否则称为无序树。假设图中为有序树，若将子结点位置互换，则变成一棵不同的树。
路径路径长度	树中两个结点之间的路径是由这两个结点之间所经过的结点序列构成的，而路径长度是路径上所经过的边的个数。
森林	森林是m(m>=0)棵互不相交的树的集合。森林的概念与树的概念十分相近，因为只要把树的根结点删去就成了森林。反之，只要给m棵独立的树加上一个结点，并把这m棵树作为该结点的子树，则森林就变成了树。

树的性质

树的结点数n等于所有结点的度数之和加1	结点的度是指结点的孩子数量，每个结点与其每个孩子都由唯一的边相连，因此树中所有结点的度数之和等于树中的边数之和。树中的结点（除根外）都由唯一的双亲，因此结点数n等于边数之和加1，即所有结点的度数之和加1。	
度为m的树中第i层上至多有m^{i-1}个结点($i>=1$)	第1层至多有1个结点（即根结点），第2层至多有m个结点，第3层至多有m^2个结点，以此类推。使用数学归纳法可推出第i层至多有m^{i-1}个结点。	
高度为h的m叉树至多有$(m^h-1)/(m-1)$个结点	当各层结点达到最大时，树中至多有$1+m+m^2+...+m^{h-1}=(m^h-1)/(m-1)$个结点。	
度为m、具有n个结点的树的最小高度为h为	为使树的高度最小，在前h-1层中，每层的结点数都要达到最大，前h-1层最多有$(m^{h-1}-1)/(m-1)$个结点。因此$(m^{h-1}-1)/(m-1)<n<=(m^h-1)/(m-1)$，即$h-1<log_m(n(m-1)+1)<=h$,可解得$h_{min}$。	
度为m、具有n个结点的树的最大高度h为n-m+1	由于树的度为m，因此至少有一个结点有m个孩子，它们处于同一层。为使树的高度最大，其他层可仅有一个结点，因此最大高度（层数）为n-m+1。由此，也可逆推出高度为h、度为m的树至少有h+m-1个结点。	