

第九章 树

树是图论中最重要的概念之一,它是基尔霍夫在解决电路理论中求解联立方程问题时首先提出的。它又是图论中结构最简单,用途最广泛的一种连通图。树与图中其他一些基本概念,如圈、割集等有着密切的联系,是图论中比较活跃的领域,是计算机中一种基本的数据结构和表示方法。现在树的概念已经越来越广泛地被应用到各个科学领域。

首先声明,本章中所指的回路均为简单回路或初级回路。

9.1 无向树

定义 9.1.1 连通无回路的无向图称为无向树,简称树,记作 T 。树中的悬挂点称为树叶,其余顶点称为分支点。每个连通分支均为树的无向图称为森林。平凡图称为平凡树。

【例 9.1.1】 图 9.1-1 中 (a) (b) 均是树, (c) 是由三棵树构成的森林。

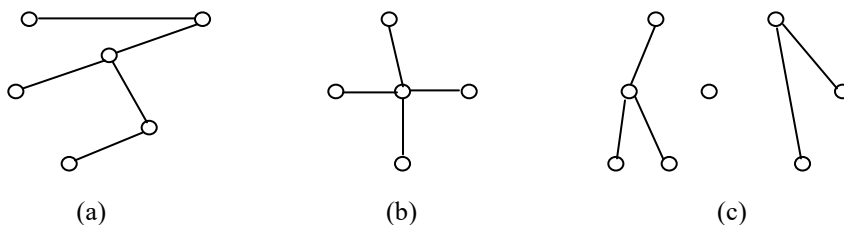


图 9.1-1 树和森林

由于树无环且无重边(否则有回路),所以树必是简单图。下面我们来讨论树的性质。

定理 9.1.1 无向图 T 是树,当且仅当以下五条之一成立。

- (1) T 中无回路且 $m = n - 1$, 其中 m 为边数, n 为顶点数。
- (2) T 是连通图且 $m = n - 1$ 。
- (3) T 中无回路,但增一条边,则得到一条且仅一条初级回路。
- (4) T 连通且每条边均是桥。
- (5) 每对顶点间有唯一的一条初级通路。

证明 (a) 由树的定义可得 (1)

施归纳于顶点数 n : 当 $n = 1$ 时, $m = 0$, 则 $m = n - 1$ 成立。

假设 当 $n = k$ 时, $m = n - 1$ 成立。则当 $n = k + 1$ 时, 因为树是连通的且无回路, 所以至少有一个度数为 1 的顶点 v , 从树中删去 v 和与它关联的边, 则得到 k 个顶点的树 T' 。根据假设它有 $k - 1$ 条边, 现将 v 和与它关联的边加到 T' 上还原成树 T , 则 T 有 $k + 1$ 个顶点, k 条边, 边数比顶点数少 1, 故 $m = n - 1$ 成立。

(b) 由 (1) 可得 (2)

反证法, 若图 T 不连通, 设 T 有 k 个连通分支 T_1, T_2, \dots, T_k ($k \geq 2$), 其顶点数分别为 n_1, n_2, \dots, n_k , 则有

$$\sum_{i=1}^k n_i = n$$

边数分别为 m_1, m_2, \dots, m_k , 则有

$$\sum_{i=1}^k m_i = m$$

因此, 有

$$m = \sum_{i=1}^k m_i = \sum_{i=1}^k (n_i - 1) = n - k < n - 1$$

即 $m < n - 1$

这与 $m = n - 1$ 矛盾, 故 T 是连通的 $m = n - 1$ 图。

(c) 由 (2) 可得 (3)

若 T 是连通图并有 $n - 1$ 条边。施归纳于顶点数 n 。

当 $n = 2$ 时, $m = n - 1 = 1$, 所以没有回路, 如果增加一条边, 只能得到唯一的一个回路。

假设 $n = k$ 时, 命题成立。

则当 $n = k + 1$ 时, 因为 T 是连通的并有 $n - 1$ 条边, 所以每个顶点都有 $d(v) \geq 1$, 并且至少有一个顶点 v_0 , 满足 $d(v_0) = 1$ 。否则, 如果每个顶点 v 都有 $d(v) \geq 2$, 那么必然会有总度数 $2m \geq 2n$, 即 $m \geq n$, 这与条件 $m = n - 1$ 矛盾。因此至少有一个顶点 v_0 , 满足 $d(v_0) = 1$ 。

删去 v_0 及其关联的边, 得到图 T' , 由假设知 T' 无回路, 现将 v_0 及其关联的边再加入到 T' , 则还原成 T , 所以 T 没有回路。

如果在连通图 T 中增加一条新边 (v_i, v_j) , 则 (v_i, v_j) 与 T 中从 v_i 到 v_j 的一条初级路径构成一个初级回路, 且该回路必定是唯一的, 否则当删去新边 (v_i, v_j) 时, T 中必有回路, 产生矛盾。

(d) 由 (3) 可得 (4)

若图 T 不连通, 则存在两个顶点 v_i 和 v_j , 在 v_i, v_j 之间没有路径, 如果增加边 (v_i, v_j) 不产生回路, 这与 (3) 矛盾, 因此 T 连通。

因为 T 中无回路, 所以删去任意一条边, 图必不连通。故图中每一条边均是桥。

(e) 由 (4) 可得 (5)

由图的连通性可知, 任意两个顶点之间都有一条通路, 是初级通路。如果这条初级通路不唯一, 则 T 中必有回路, 删去回路上的任意一条边, 图仍连通, 与 (4) 矛盾。故任意两个顶点之间有唯一一条初级回路。

(f) 由 (5) 可得树的定义

每对顶点之间有唯一一条初级通路, 那么 T 必连通, 若有回路, 则回路上任意两个顶点之间会有两条初级通路, 与 (5) 矛盾。故图连通且无回路, 是树。 证毕

定理 9.1.2 任何一棵非平凡树 T 至少有两片树叶。

证明 设 T 是 (n, m) 图, $n \geq 2$, 有 k 片树叶, 其余顶点度数均大于或等于 2。则

$$\sum_{i=1}^n d(v_i) \geq 2(n - k) + k = 2n - k$$

而

$$\sum_{i=1}^n d(v_i) = 2m = 2(n - 1) = 2n - 2$$

所以 $2n - 2 \geq 2n - k$ 即 $k \geq 2$ 。 证毕

【例 9.1.2】 T 是一棵树, 有两个顶点度数为 2, 一个顶点度数为 3, 三个顶点度数为 4, T 有几片树叶?

解 设树 T 有 x 片树叶。则 T 的顶点数

$$n = 2 + 1 + 3 + x$$

T 的边数

$$m = n - 1 = 5 + x$$

又由握手定理

$$2m = \sum_{i=1}^n d(v_i)$$

得

$$2(5 + x) = 2 \cdot 2 + 3 \cdot 1 + 4 \cdot 3 + x$$

所以 $x = 9$ 即 树 T 有 9 片树叶。

请读者自己画出两棵具有上述度数列的非同构的树。

【例 9.1.3】 画出所有非同构的七阶无向树

解 设 T_i 是七阶无向树。因为 $n = 7$ ，所以 $m = 6$ ，又因为 $2m = \sum_{i=1}^n d(v_i)$ ，所以七个顶点分

配 12 度，且由树是连通简单图知 $1 \leq d(v) \leq 6$ 。则 T_i 的度数列必是下列情况之一。

- (1) 1, 1, 2, 2, 2, 2, 2
- (2) 1, 1, 1, 2, 2, 2, 3
- (3) 1, 1, 1, 1, 2, 2, 4
- (4) 1, 1, 1, 1, 2, 3, 3
- (5) 1, 1, 1, 1, 1, 2, 5
- (6) 1, 1, 1, 1, 1, 3, 4
- (7) 1, 1, 1, 1, 1, 1, 6

注意到，不同构的度数列对应不同的树，但对应同一度数列的非同构的树不一定唯一，所以对应 (1) 有 T_1 ，对应 (2) 的有 T_2 、 T_3 和 T_4 ，对应 (3) 的有 T_5 和 T_6 ，对应 (4) 的有 T_7 和 T_8 ，对应 (5) 的有 T_9 ，对应 (6) 的有 T_{10} ，对应 (7) 的有 T_{11} （见图 9.1-2）。

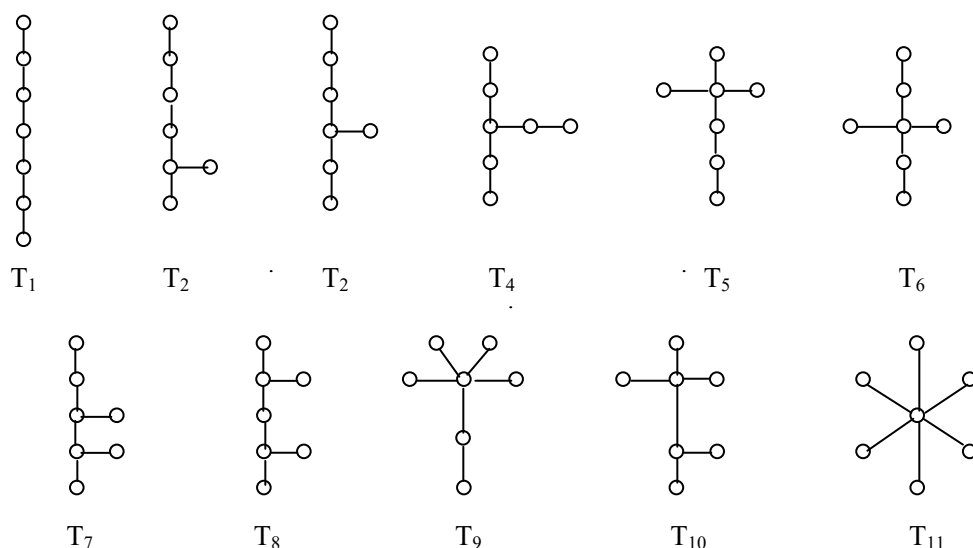


图 9.1-2 不同构的七阶无向树

生成树

有一些图，本身不是树，但它的某些子图却是树，其中很重要的一类是生成树。

定义 9.1.2 若无向图 G 的一个生成子图 T 是树，则称 T 是 G 的一棵生成树。

如果 T 是 G 的一棵生成树，则称 G 在 T 中的边为 T 的**树枝**， G 不在 T 中的边为 T 的**弦**， T 的所有的弦的集合的导出子图称 T 的**余树**。易知，余树不一定是树，更不一定是生成树。

例如图 9.1-3 中， T_1 和 T_2 是图 G 的两棵生成树， \bar{T}_1 和 \bar{T}_2 是分别对应于它们的余树。

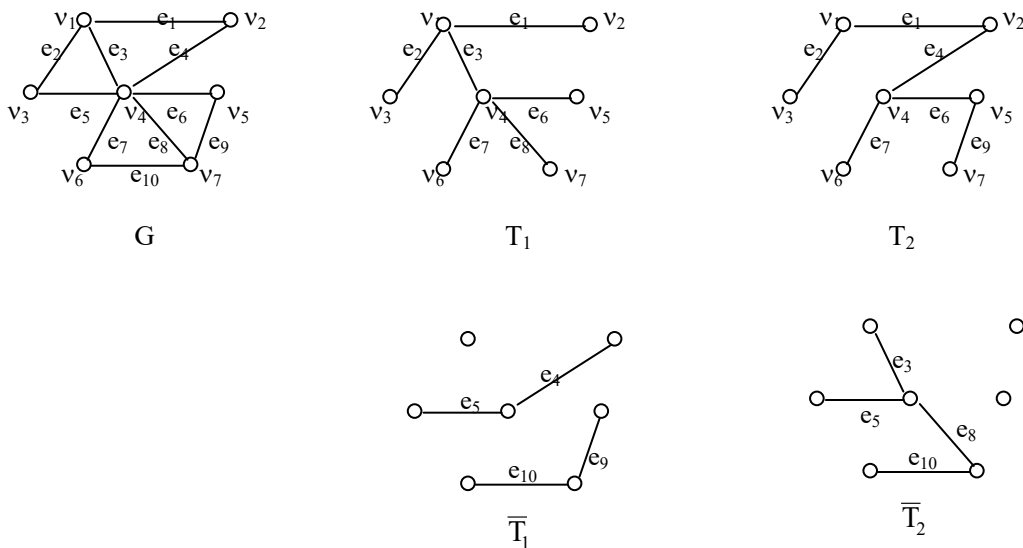


图 9.1-3 图的生成树和余树

由图 9.1-3 可见, G 与 T_1 、 T_2 的区别是 G 中有回路, 而它的生成树无回路, 因此要在一个连通图 G 中找到一棵生成树, 只要不断的从 G 的回路上删去一条边, 最后所得无回路的子图就是 G 的一棵生成树。于是有

定理 9.1.3 无向图 G 有生成树的充分必要条件是 G 为连通图。

证明 必要性, 反证法

若 G 不连通, 则它的任何生成子图也不连通, 因此不可能有生成树, 与 G 有生成树矛盾, 故 G 是连通图。

充分性

设 G 连通, 则 G 必有连通的生成子图, 令 T 是 G 的含有边数最少的生成子图, 于是 T 中必无回路 (否则删去回路上的一条边不影响连通性, 与 T 含边数最少矛盾), 故 T 是一棵树, 即生成树。证毕

推论 1 设 G 是 (n, m) 无向连通图, 则 $m \geq n-1$ 。

推论 2 设 G 是 (n, m) 无向连通图, T 是 G 的一棵生成树, 则 T 的余树 \bar{T} 中含 $m-n+1$ 条边。

推论 3 设 T 是连通图 G 的一棵生成树, \bar{T} 为 T 的余树, C 为 G 中任意一个圈, 则 $E(\bar{T}) \cap E(C) \neq \emptyset$ 。

证明 若 $E(\bar{T}) \cap E(C) = \emptyset$, 则 $E(C) \subseteq E(T)$, 说明 C 是 T 中的一个圈, 与 T 为树矛盾。证毕

一般情况, 图 G 的生成树不是唯一的, 这里的不唯一有两个含义, 一种是标定图的生成树所含的树枝不同, 另一种是不考虑标定的生成树是不同构的。但无论如何, 一个连通的 (n, m) 图, 其任意一棵生成树的树枝数一定是 $n-1$, 从而弦数也是固定不变的 $m-n+1$ 。

由树的性质可知, 对于图 G 的一棵生成树 T , 每增加一条弦, 便得到一条初级回路。

例如在图 9.1-3 所示的 T_1 中:

加弦 e_5 , 得初级回路 $e_2e_3e_5$;

加弦 e_4 , 得初级回路 $e_1e_3e_4$;

加弦 e_9 , 得初级回路 $e_6e_8e_9$;

加弦 e_{10} , 得初级回路 $e_7e_8e_{10}$ 。

同样，在图 9.1-3 所示的 T_2 中，分别加弦 e_3, e_5, e_8, e_{10} ，则分别产生初级回路 $e_1e_3e_4, e_1e_4e_5e_2, e_6e_8e_9, e_7e_6e_9e_{10}$ 。

这些初级回路有一个共同特点：它们中均只含一条弦，其余的边均是树枝，我们称这样的回路为**基本回路**。对于 G 的每棵生成树 T ， $m-n+1$ 条弦对应着 $m-n+1$ 个基本回路，这些基本回路构成的集合称为对应 T 的**基本回路系统**。显然不同的生成树对应不同的基本回路系统。如图 9.1-3 中， G 对应 T_1 的基本回路系统为 $\{e_2e_3e_5, e_1e_3e_4, e_6e_8e_9, e_7e_8e_{10}\}$ ， G 对应 T_2 的基本回路系统为 $\{e_1e_3e_4, e_1e_4e_5e_2, e_6e_8e_9, e_7e_6e_9e_{10}\}$ 。但是它们所含的元素个数均是 $m-n+1=4$ 。在网络分析中，基本回路具有重要意义。

另一方面，从树 T 中删去一边，便将 T 分成两棵树，即两个连通分支，图 G 中连接这两个连通分支的边的集合，称为对应于这条边的**基本（边）割集**。

例如在图 9.1-3 中的 G 和 T_1

对应树枝 e_1 ，有基本割集 $\{e_1, e_4\}$ ；

对应树枝 e_2 ，有基本割集 $\{e_2, e_5\}$ ；

对应树枝 e_3 ，有基本割集 $\{e_3, e_4, e_5\}$ ；

对应树枝 e_6 ，有基本割集 $\{e_6, e_9\}$ ；

对应树枝 e_7 ，有基本割集 $\{e_7, e_{10}\}$ ；

对应树枝 e_8 ，有基本割集 $\{e_8, e_9, e_{10}\}$ ；

同样对于图 9.1-3 中的 G 和 T_2 ，对应树枝 $e_1, e_2, e_4, e_6, e_7, e_9$ ，分别有基本割集： $\{e_1, e_3, e_5\}, \{e_2, e_5\}, \{e_4, e_3, e_5\}, \{e_6, e_8, e_{10}\}, \{e_7, e_{10}\}, \{e_9, e_{10}\}$ 。

这些割集所具有的共同特点是：每个割集中均只含生成树的一个树枝，其余的均是弦。对于图 G 的任何一棵生成树 T ，对应每个树枝的基本割集所构成的集合，称为对应 T 的**基本割集系统**。如图 9.1-3 中， G 对应 T_2 的基本割集系统为 $\{\{e_1, e_3, e_5\}, \{e_2, e_5\}, \{e_4, e_3, e_5\}, \{e_6, e_8, e_{10}\}, \{e_7, e_{10}\}, \{e_9, e_{10}\}\}$ ，显然不同的生成树有不同的基本割集系统，但其元素个数均为 $n-1$ 。

最小生成树

带权图的生成树是实际应用较多的树，一个很实际的问题是：假设你是一个设计师，欲架设连接 n 个村镇的电话线，每个村镇设一个交换站。已知由 i 村到 j 村的线路 $e = (v_i, v_j)$ 造价为 $\omega(e) = \omega_{ij}$ ，要保证任意两个村镇之间均可通话，请设计一个方案，使总造价最低。

这个问题的数学模型为：在已知的带权图上求权最小的生成树。

定义 9.1.3 设无向连通带权图 $G = \langle V, E, \omega \rangle$ ， G 中权最小的生成树称为 G 的最小生成树（最优树）。

定理 9.1.4 设连通图 G 的各边的权均不相同，则回路中权最大的边必不在 G 的最小生成树中。

（证明略）

定理的结论是显然的，由此寻找带权图 G 的最小生成树，可以采用破圈法，即在图 G 中不断去掉回路中权最大的边。

求最小生成树的另一个更有效率的算法是克鲁斯卡尔（Kruskal）的避圈法：

- （1）选 $e_1 \in E(G)$ ，使得 $\omega(e_1) = \min$ 。
- （2）若 e_1, e_2, \dots, e_i 已选好，则从 $E(G) - \{e_1, e_2, \dots, e_i\}$ 中选取 e_{i+1} ，使得
 - （i） $G[\{e_1, e_2, \dots, e_i\}]$ 中无圈，且
 - （ii） $\omega(e_{i+1}) = \min$ 。
- （3）继续进行到选得 e_{n-1} 为止。

【例 9.1.4】 求下面带权图（图 9.1-4（a））的最小生成树。（它的实际背景是北京与巴黎、纽约、东京、伦敦、墨西哥城这六大城市间的航空路线距离图，单位是百公里。）

解 图 9.1-4（b）中实线是用破圈法得到的最小生成树（虚线是去掉的回路上的边），（c）中粗实线是用避圈法得到的最小生成树。其权均为 122。

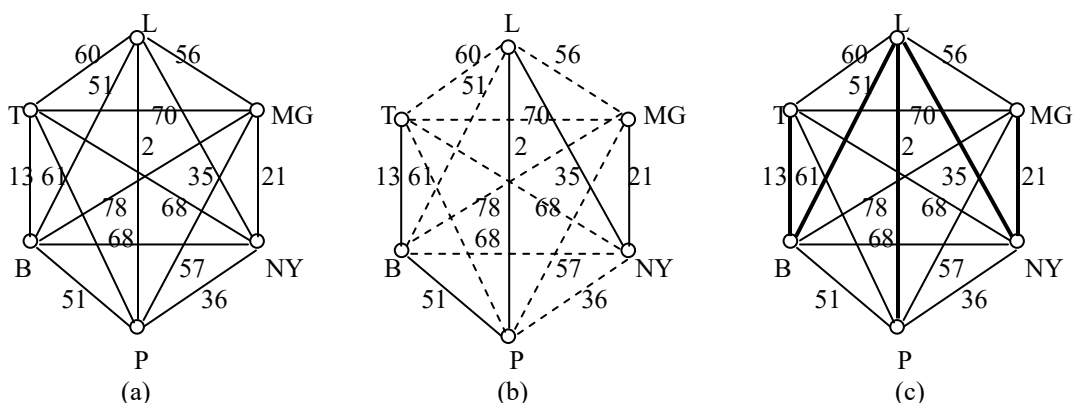


图 9.1-4 用破圈法和避圈法得到的最小生成树

事实上因为边（L，B）与（P，B）的权相等，所以最小生成树并不唯一甚至于并不同构（如图 9.1-4 的（b）与（c）），但是它们的权是相等的。

9.2 根树及其应用

有向树 在不考虑方向时为无向树的有向图。根树是一种特殊的有向树。

定义 9.2.1 T 是一棵有向树，若 T 中恰有一个顶点入度为 0，其余顶点的入度均为 1，则称 T 为**根树**。入度为 0 的顶点称**树根**，出度为 0 的顶点称**树叶**，入度为 1、出度不为 0 的顶点称**内点**，内点和树根统称为**分支点**。

例如图 9.2-1 中的（a）、（b）、（c）均是有向树，但只有（c）是根树。

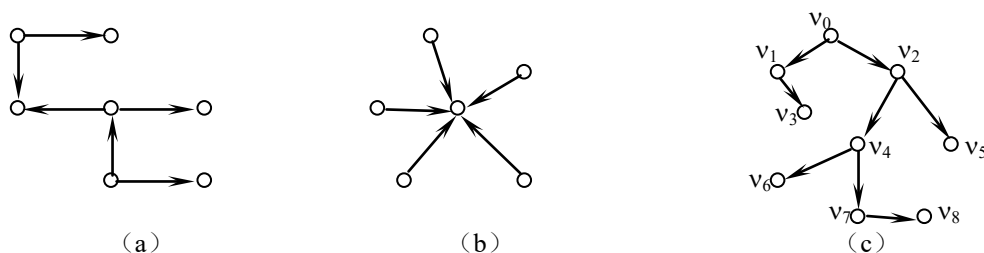


图 9.2-1 有向树和根树

由定义和例题易知，对于 (n, m) 有向树，同样有 $m = n - 1$ ，且根树一定是有向树，但反之未必。

在根树中，从树根 v_0 到每个顶点 v_i 有唯一一条初级通路，该通路的长度称为点 v_i 的**层数**，记作 $\ell(v_i)$ ，其中最大的层数称为**树高**，记作 $h(T)$ 。

例如图 9.2-1（c）中， $\ell(v_0) = 0$ ， $\ell(v_1) = 1$ ， $\ell(v_4) = 2$ ， $\ell(v_8) = 4 = h(T)$ 。

习惯上将根树画成树根在上，各边箭头均朝下的形状（如（c）），并为方便起见，略去各边上的箭头，可以看出，根树上的各个顶点有了层次关系。

一棵根树常常被形象的比作一棵家族树，如果顶点 u 邻接到顶点 v ，则称 u 为 v 的父亲， v 为 u 的儿子；共有同一个父亲的顶点称为兄弟；如果顶点 u 可达顶点 v ，则称 u 是 v 的祖先， v

是 u 的后代。在根树 T 中，所有的内点、树叶均是树根的后代，由某个顶点 v_i 及其所有的后代构成的导出子图 T' 称为 T 的以 v_i 为根的子根树。

例如图 9.2-1 (c) 中， v_3 是 v_1 的儿子， v_1 是 v_3 的父亲； v_4 与 v_5 是兄弟， $T' = G[\{v_2, v_4, v_5, v_6, v_7, v_8\}]$ 是以 v_2 为根的 T 的**子根树**。

定义 9.2.2 在根树 T 中，如果每一层的顶点都按一定的次序排列，则称 T 为有序树。

在画有序树时，常假定每一层的顶点是按从左到右排序的。

例如图 9.2-2 中的 (a) 和 (b) 表示的是不同的有序树。而如果不考虑同层顶点的次序，则 (a) 和 (b) 表示的是同一棵根树。



图 9.2-2 有序树

【例 9.2.1】 英语句子 “The big elephant ate the peanut” 可以图解为图 9.2-3，称之为这个英语句子的语法树。可见，句子的语法树是一棵有序树。

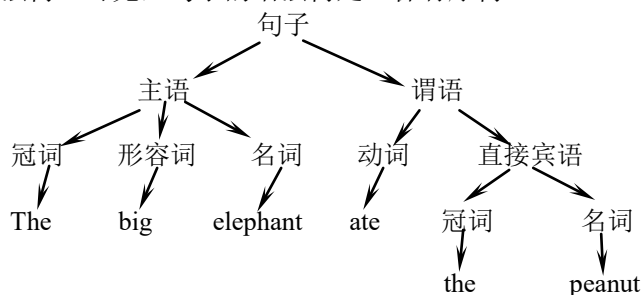


图 9.2-3 语法树

定义 9.2.3 设 T 是一棵根树

- (1) 若 T 的每个顶点至多有 m 个儿子，则称 T 为 m 元树。
- (2) 若 T 的每个顶点都有 m 个或 0 个儿子，则称 T 为 m 元正则树。
- (3) 若 T 是 m 元树，并且是有序的，则称 T 为 m 元有序树。
- (4) 若 T 是 m 元正则树，并且是有序的，则称 T 为 m 元有序正则树。
- (5) 若 T 是 m 元正则树，且所有树叶的层数都等于树高，则称 T 为 m 元完全正则树。
- (6) 若 T 是 m 元完全正则树，且是有序的，则称 T 为 m 元有序完全正则树。
- (7) 设 T 是 m 元树，如果为 T 中每个顶点的儿子规定了确定的位置，则称 T 为 m 元位置树。

例如图 9.2-4 中，(a) 和 (b) 可看成相等的二元有序树，但是不是相同的二元位置树，(c) 是二元正则树，(d) 是二元完全正则树。

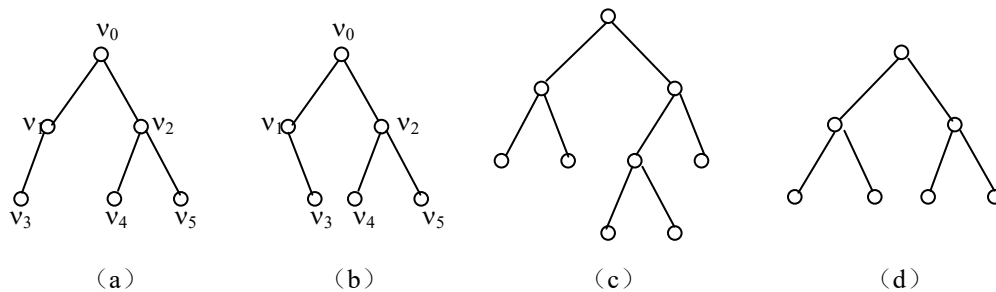


图 9.2-4 根树的分类

在所有的 m 元树中，二元树居重要地位，其中二元有序正则树应用最为广泛。在二元有序正则树中，以分支点的两个儿子分别作为树根的两棵子树通常称为该分支点的左子树和右子树。

【例 9.2.2】 用二元树（图 9.2-5）表示算术表达式： $(a-b) \div (c \times d + e)$

对计算机来说，二元位置树最容易处理，因此常将有序树转化为二元位置树，步骤如下：

- (1) 对于每个顶点只保留左儿子。
- (2) 兄弟间从左到右连接。
- (3) 对于每个分支点，保留的左儿子仍做左儿子，右边邻接的顶点做右儿子。

例如图 9.2-6 (a) 是棵有序树，(b) 是表示 (a) 的一棵二元树，(c) 是相应的二元位置树。

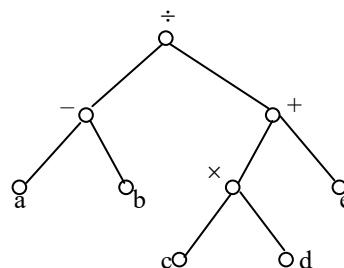


图 9.2-5 算术树

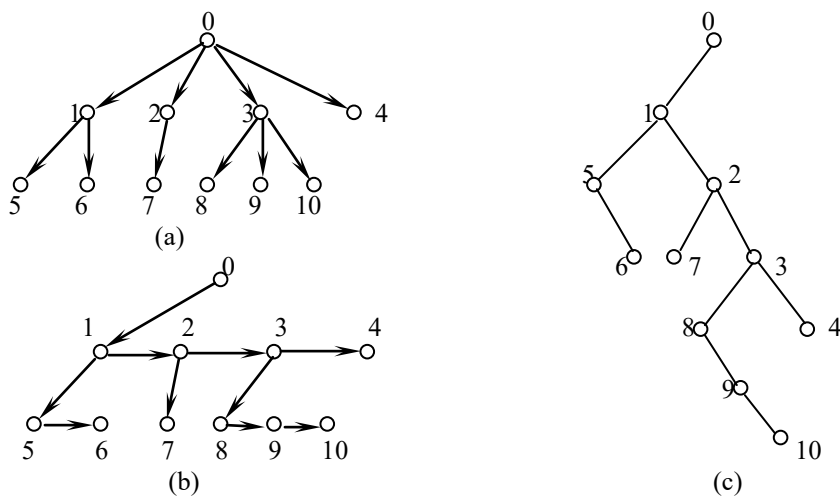


图 9.2-6 有序树转化为二元位置树

此方法可以推广到有序森林上去，只是将森林中每棵树的根看作兄弟。

例如图 9.2-7 (a) 是有序森林，(b) 是表示 (a) 的一棵二元树，(c) 是相应的二元位置树。

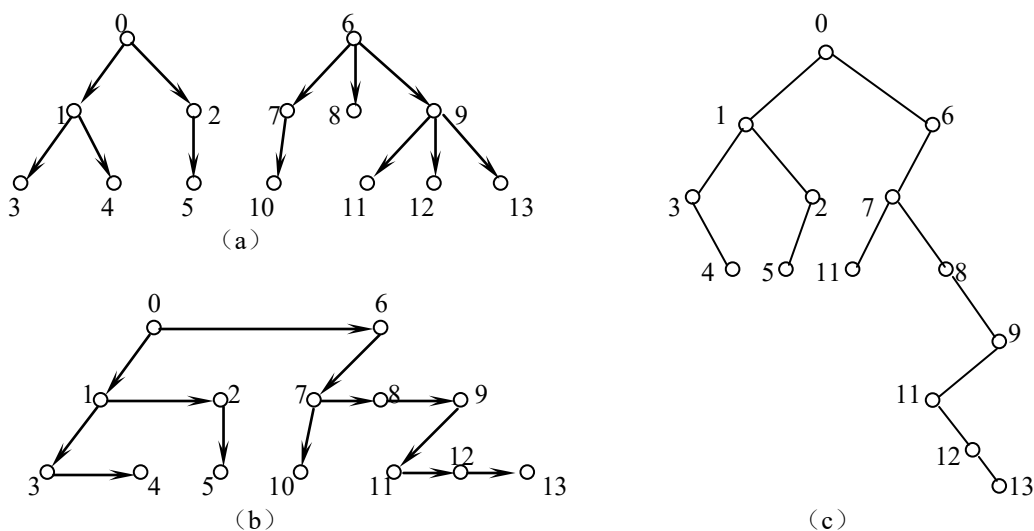


图 9.2-7 森林转化为二元位置树

显然这种方法是可逆的，即任何一棵二元位置树，也可以转化为有序树或有序森林。

最优树

下面提到的树均是正则树。

定义 9.2.4 设根树 T 有 t 片树叶 v_1, v_2, \dots, v_t ，它们分别带权 $\omega_1, \omega_2, \dots, \omega_t$ ，则称 T 为 (叶) 带权树，称 $W(T) = \sum_{i=1}^t \omega_i \ell_i$ 为 T 的权，其中 ℓ_i 是 v_i 的层数。

【例 9.2.3】 求 4 片树叶分别带权 5,6,7,12 的二元树的权。

解 见图 9.2-8，根据题意，我们构造出了四棵树叶具有不同权的带权二元树：

(a) $W(T_1) = 61$, (b) $W(T_2) = 74$, (c) $W(T_3) = 59$, (d) $W(T_4) = 60$ 。

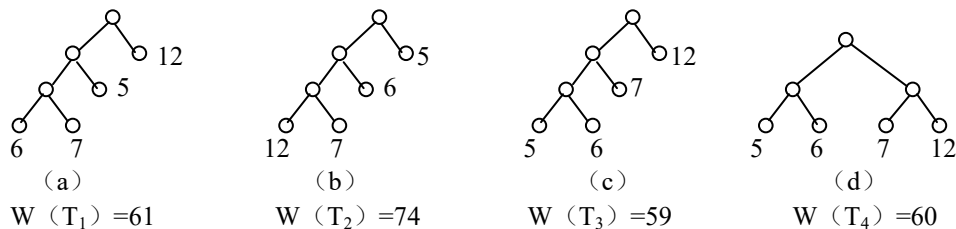


图 9.2-8 带权二元树

由于树叶的层数不同，叶权也大小各异，因此树权是不同的，但其中必存在一棵权最小的二元树。

定义 9.2.5 在所有叶带权 $\omega_1, \omega_2, \dots, \omega_t$ 的二元树中，权最小的二元树称最优二元树，简称最优树。

如何寻求最优二元树？1952 年哈夫曼 (Huffman) 给出了求最优二元树的算法，用该算法求得的最优树又称 Huffman 树。

Huffman 算法:

令 $S = \{\omega_1, \omega_2, \dots, \omega_t\}$, $\omega_1 \leq \omega_2 \leq \dots \leq \omega_t$, ω_i 是树叶 v_i 所带的权 ($i = 1, 2, \dots, t$)。

(1) 在 S 中选取两个最小的权 ω_i, ω_j , 使它们对应的顶点 v_i, v_j 做兄弟, 得一支点 v_r , 令其带权 $\omega_r = \omega_i + \omega_j$ 。

(2) 从 S 中去掉 ω_i, ω_j , 再加入 ω_r 。

(3) 若 S 中只有一个元素, 则停止, 否则转到 (1)。

【例 9.2.4】 构造一棵带权 1,3,3,4,6,9,10 的最优二元树, 并求其权 $W(T)$ 。

解 构造过程如图 9.2-9 所示。

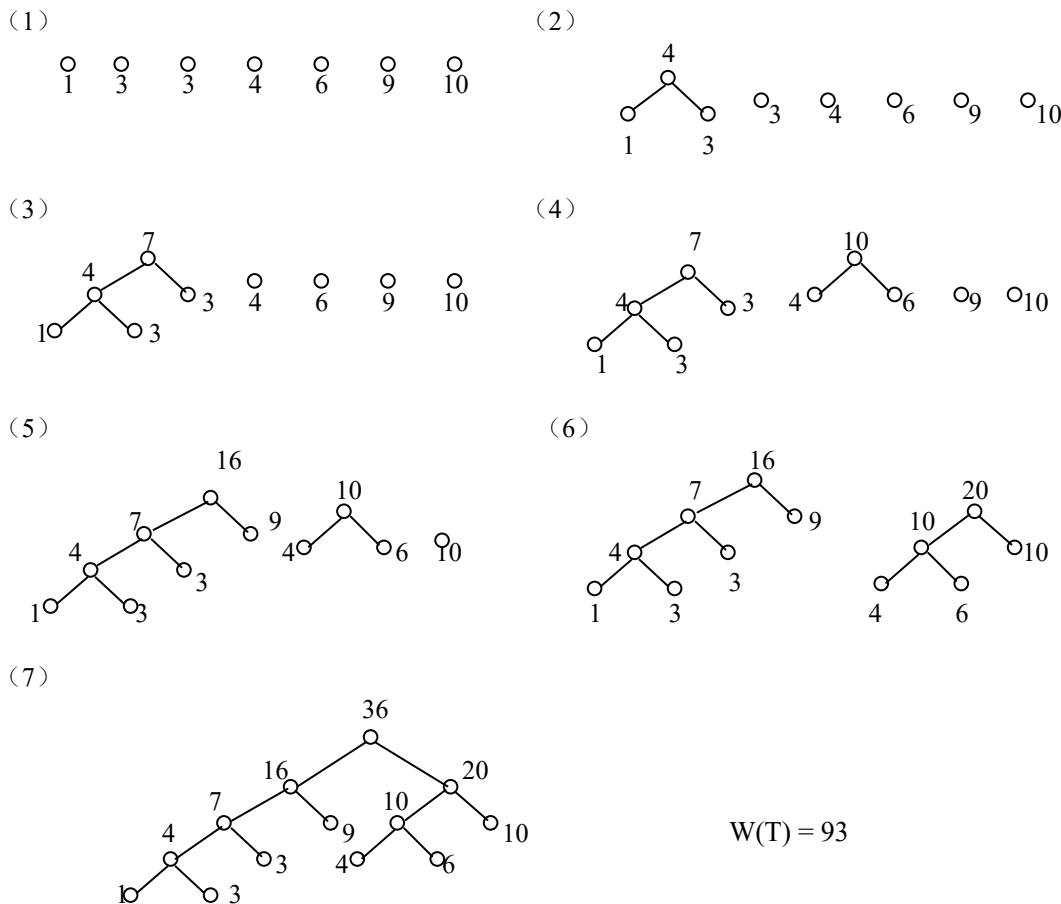


图 9.2-9 Huffman 最优树的构造过程

一般来说, 带权 $\omega_1, \omega_2, \dots, \omega_t$ 的最优二元树并不唯一。

下面证明 Huffman 算法的正确性, 先证下面的引理。

引理 设 T 是一棵带权 $\omega_1 \leq \omega_2 \leq \dots \leq \omega_t$ 的最优二元树, 则带最小权 ω_1, ω_2 的树叶 v_1 和 v_2 是兄弟, 且以它们为儿子的分支点层数最大。

证明 设 v 是 T 中离根最远的分支点, 它的两个儿子 v_a 和 v_b 都是树叶, 分别带权 ω_a 和 ω_b , 而不是 ω_1 和 ω_2 。并且从根到 v_a 和 v_b 的通路长度分别是 ℓ_a 和 ℓ_b , $\ell_a = \ell_b$ 。故有

$$\ell_a \geq \ell_1, \ell_b \geq \ell_2$$

现在将 ω_a 和 ω_b 分别与 ω_1 和 ω_2 交换, 得一棵新的二元树记为 T' , 则

$$W(T) = \omega_1 \ell_1 + \omega_2 \ell_2 + \dots + \omega_a \ell_a + \omega_b \ell_b + \dots$$

$$W(T') = \omega_a \ell_1 + \omega_b \ell_2 + \dots + \omega_1 \ell_a + \omega_2 \ell_b + \dots$$

于是

$$W(T) - W(T') = (\omega_1 - \omega_a)(\ell_1 - \ell_a) + (\omega_2 - \omega_b)(\ell_2 - \ell_b) \geq 0$$

即 $W(T) \geq W(T')$ ，又 T 是带权 $\omega_1, \omega_2, \dots, \omega_t$ 的最优树，应有 $W(T) \leq W(T')$ ，因此 $W(T) = W(T')$ 。从而可知 T' 是将权 ω_1, ω_2 与 ω_a, ω_b 对调得到的最优树，故而 $\ell_a = \ell_1, \ell_b = \ell_2$ ，即带权 ω_1 和 ω_2 的树叶是兄弟，且以它们为儿子的分支点层数最大。证毕

定理 9.2.1 (Huffman 定理) 设 T 是带权 $\omega_1 \leq \omega_2 \leq \dots \leq \omega_t$ 的最优二元树，如果将 T 中带权为 ω_1 和 ω_2 的树叶去掉，并以它们的父亲作树叶，且带权 $\omega_1 + \omega_2$ ，记所得新树为 \hat{T} ，则 \hat{T} 是带权为 $\omega_1 + \omega_2, \omega_3, \dots, \omega_t$ 的最优树。

证明 由题意可知

$$W(T) = W(\hat{T}) + \omega_1 + \omega_2$$

若 \hat{T} 不是最优树，则必有另一棵带权 $\omega_1 + \omega_2, \omega_3, \dots, \omega_t$ 的最优树 \tilde{T} 。令 \tilde{T} 中带权 $\omega_1 + \omega_2$ 的树叶生出两个儿子，分别带权 ω_1 和 ω_2 ，得到新树 T' ，则

$$W(T') = W(\tilde{T}) + \omega_1 + \omega_2$$

因为 \tilde{T} 是带权为 $\omega_1 + \omega_2, \omega_3, \dots, \omega_t$ 的最优树，所以

$$W(\tilde{T}) \leq W(\hat{T})$$

如果 $W(\tilde{T}) < W(\hat{T})$ ，则必有 $W(T') < W(T)$ ，与 T 是带权 $\omega_1 \leq \omega_2 \leq \dots \leq \omega_t$ 的最优二元树矛盾，故 \hat{T} 是带权为 $\omega_1 + \omega_2, \omega_3, \dots, \omega_t$ 的最优树。证毕

由 Huffman 定理易知 Huffman 算法的正确性。

最优树的一个直接用处是前缀码的设计。

在远程通讯中，通常的电报是用 0 和 1 组成的序列来表示英文字母和标点符号的，英文有 26 个字母，所以只要用长度为 5 的等长字符串就可表达不同的字母了。发送端只要发送一条 0 和 1 组成的字符串，它正好是信息中字母对应的字符序列。在接收端，将这一长串字符分成长度为 5 的序列就得到了相应的信息。

但是字母在信息中出现的频繁程度是不一样的，例如字母 e 和 t 在单词中出现的频率要远远大于字母 q 和 z 在单词中出现的频率。因此人们希望能用较短的字符串表示出现较频繁的字母，这样就可缩短信息字符串的总长度，显然如能实现这一想法是很有价值的。对于发送端来说，发送长度不同的字符串并无困难，但在接收端，怎样才能准确无误地将收到的一长串字符分割成长度不一的序列，亦即接收端如何译码呢？例如若用 00 表示 t，用 01 表示 e，用 0001 表示 y，那么当接收到字符串 0001 时，如何判断信息是 te 还是 y 呢？为了解决这个问题，先引入前缀码的概念。

定义 9.2.6 设 $a_1 a_2 \dots a_n$ 是长度为 n 的符号串，称其子串 $a_1, a_1 a_2, \dots, a_1 a_2 \dots a_{n-1}$ 分别为该符号串的长度为 1, 2, $\dots, n-1$ 的前缀。

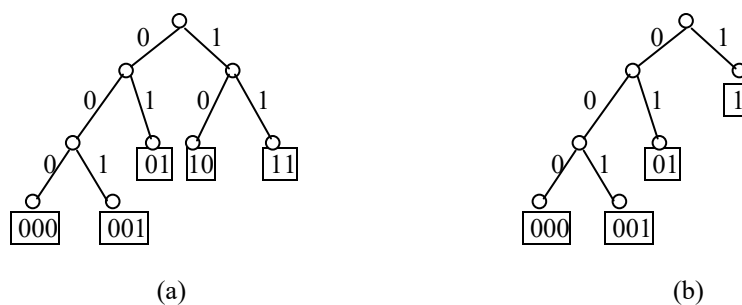
设 $A = \{\beta_1, \beta_2, \dots, \beta_n\}$ 为一个符号串集合，若 A 中任意两个不同的符号串 β_i 和 β_j 互不为前缀，则称 A 为一组**前缀码**，若符号串中只出现两个符号，则称 A 为二元前缀码。

如 $\{0, 10, 110, 1110, 1111\}$ 是前缀码， $\{00, 001, 011\}$ 不是前缀码。

二元前缀码与正则二元树有一一对应关系。

在一棵正则二元树中，将每个顶点和它的左儿子之间的边标记为 0；和它的右儿子之间的边标记为 1，如图 9.2-10 (a)，把从根到每个树叶所经过的边的标记序列作为树叶的标记，由于每片树叶的标记的前缀是它的祖先的标记，而不可能是任何其他树叶的标记，所以这些树叶的标记就是前缀码。由图 9.2-10 (a) 可看出前缀码是 $\{000, 001, 01, 10, 11\}$ 。

相反，如果给定前缀码，也可找出对应的二元树。例如，前缀码 $\{000, 001, 01, 1\}$ 对应的二元树如图 9.2-10 (b) 所示。



当知道了要传输的符号的频率时，可用各个符号出现的频率作权，用 Huffman 算法求一棵最优树 T，由 T 产生的前缀码称为最佳前缀码（又称 Huffman 码），用这样的前缀码传输对应的符号可以使传输的二进制码最省。

0: 20%; 1: 15%; 2: 10%; 3: 10%; 4: 10%;
5: 5%; 6: 10%; 7: 5%; 8: 10%; 9: 5%.

解 (1) 令 i 对应叶权 ω_i , $\omega_i = 100i$, 则

构造一棵带权 5,5,5,10,10,10,10,10,15,20 的最优二元树（见图 9.2-11），数字与前缀码的对应关系见图右侧。

图 9.2-11 最优二元树与最佳前缀码

$$(2) (2 \times 20\% + 3 (10\% + 15\% + 10\% + 10\%) + 4 (5\% + 10\% + 10\%) + 5 (5\% + 5\%)) \times 10000 = 32500$$

(3) 因为用等长码传输十个数字码长为 4, 即用等长的码传 10000 个数字需 40000 个二进制码, 故用最佳前缀码传节省了 7500 个二进制码。

【例 9.3.1】 判断下列命题是否为真。

- (1) 若 n 阶无向简单图 G 的边数 $m = n-1$, 则 G 一定是树。
- (2) 若有向图 G 仅有一个顶点的入度为 0, 其余顶点的入度都是 1, 则 G 一定是有向树。
- (3) 任何树 T 都至少有两片树叶。
- (4) 任何无向图 G 都至少有一棵生成树。
- (5) 根树中最长初级通路的端点都是树叶。
- (6) 无向连通图 G 中的任何一条边都可以成为 G 的某棵生成树的树枝。
- (7) (n, m) 图的每棵生成树都有 $n-1$ 条树枝。
- (8) 以 1, 1, 1, 1, 2, 2, 4 为度数列的非同构的树有 2 棵。
- (9) 任何无向树的边均是桥。
- (10) G 是 (n, m) 无向图, 若 $m \geq n$, 则 G 中必有圈。
- (11) G 是 (n, m) 无向连通图, 若要求 G 的一棵生成树, 必删去 G 中的 $m-n+1$ 条边。

解答与分析

(1) 假命题。当 G 是非连通无向图时, 顶点数和边数可能满足 $m = n-1$, 但 G 不是树 (如图 9.3-1 所示)。

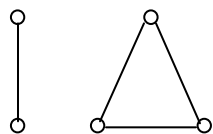


图 9.3-1

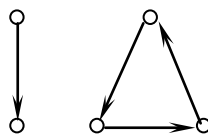


图 9.3-2

(2) 假命题。当 G 是非连通的有向图时, 各点入度可能满足命题条件, 但 G 不是有向树 (如图 9.3-2 所示)。

(3) 假命题。在平凡树中没有两片树叶。

(4) 假命题。当且仅当 G 是连通图时, G 中才有生成树。例如图 9.3-1 所示非连通图没有生成树。

(5) 假命题。根树中最长初级通路的两个端点, 一个是树叶, 而另一个是树根。

(6) 假命题。无向连通图 G 中若有环, 则环永远无法成为生成树的树枝, 因为树中无圈。

(7) 真命题。因为 (n, m) 图的每棵生成树都是其生成子图, 必有 n 个顶点。

(8) 真命题。如图 9.3-3 所示, 在一棵中两个 2 度顶点邻接, 在另一棵中两个 2 度顶点不邻接。

(9) 真命题。由树的等价定义可知。

(10) 真命题。若 G 是非简单图, 则 G 中必有环或平行边, 故 G 中有圈。

若 G 是简单连通图, 则 G 中必有圈。否则 G 是树, $m = n-1$, 这与 $m \geq n$ 矛盾。

若 G 是简单非连通图, 设其有 k 个连通分支, 若每个连通分支均无圈, 则 G 是森林, 必有 $m = n-k$, 与 $m \geq n$ 矛盾, 故至少有一个连通分支不是树, 在此分支中有圈。

(11) 真命题。因每棵生成树必有 $n-1$ 条边。



图 9.3-3

【例 9.3.2】 G 是 n ($n \geq 3$) 阶连通无向图, 只有一个一度顶点, 证明 G 中必有圈。

分析 $n \geq 3$ 说明 G 不是平凡图, 只有一个一度顶点, 说明 G 不是树, 因此, 无向图为树的两个条件至少有一个不满足。

证明 因为非平凡图 G 只有一个一度顶点, 所以 G 不是树, 又因为 G 是连通图, 因此 G 中必有回路。结论成立。

【例 9.3.3】 (1) 若 T 是高度为 k 的二元树, 则 T 的树叶数最多为 2^k 。(2) 若 T 是高度为 k 的二元完全正则树, 则 T 的顶点数为 $2^{k+1}-1$ 。

分析 (1) 在同样高度的二元树中, 二元完全正则树的树叶最多, 只需证明二元完全正则树的树叶数是 2^k 片。(2) 利用 (1) 的结果。

证明 (1) T 是高度为 k 的二元完全正则树, 对 k 做归纳:

当 $k=1$ 时, 树叶数 $t=2=2^1$, 结论成立。

假设 $k=n$ 时结论成立, $t=2^n$ 。

当 $k=n+1$ 时, 因为在原来 2^n 片树叶中, 每片叶均增加了 2 个儿子变成分支点, 所以有树叶 $t=2 \times 2^n = 2^{n+1}$, 由于在同样高度的二元树中, 二元完全正则树的树叶最多, 故结论成立。

(2) 利用 (1) 知, 在二元完全正则树中, 树叶数亦即高度为 k 的顶点有 2^k 个, 故树高为 k 的顶点总数为 $1+2+2^2+\cdots+2^k = \frac{1 \times (2^{k+1}-1)}{2-1} = 2^{k+1}-1$ 。

【例 9.3.4】 若图 $G = \langle V, E \rangle$ 连通且 $e \in E$, 证明:

(1) e 属于每一个生成树的充要条件是 e 为 G 的割边。

(2) e 不属于 G 的任一个生成树的充要条件是 e 为 G 中的环。

分析 本题 (1), (2) 均需从充分和必要两部分予以论证。在 (1) 中如 e 属于每个生成树, 需证 G 中删去 e 后必不连通, 否则 e 必不属于某个生成树与题设矛盾。在 (2) 中要证明 e 是环, 可证 e 是仅含其本身的一个回路, 否则 e 必属于某棵生成树。

证明 (1) 充分性 设 e 属于 G 中每个生成树 T , 若 e 不是割边, 则 $G-e$ 连通, 因此在 $G-e$ 中必存在生成树 T , 因为 $V(G-e) = V(G)$, 所以 T 也是 G 中的生成树。但 T 中不包含 e , 与题设矛盾。

必要性 设 e 是 G 的割边, 若有 G 的某个生成树 T 不包含 e , 则 $T+e$ 必包含回路 C , 且 $e \in C$ 。在 C 中删去 e 后仍连通, 故与 e 是割边的假设矛盾。

(2) 充分性 因为 G 连通, 必有生成树 T , 设 $e \notin T$, 则 $T+e$ 包含回路 C , 如果 C 中除 e 外另有边 $e_1 \in G$, 构造 $T' = T+e-e_1$, T' 必仍连通, 因为 $|E(T')| = |E(T)| = m-1$, 故 T' 也是 G 的一棵生成树, 但 T' 包含 e , 与题设矛盾。故 C 中没有异于 e 的边, 即 e 是 G 中的环。

必要性 若 e 是 G 中的环, 则 e 不能属于 G 的任一生成树, 因为树是连通无回路的。

【例 9.3.5】 证明: 连通图中任一回路 C 与任一边割集 S 有偶数 (包括 0) 条公共边。

证明 假设边割集 S 将连通图 G 分成两个连通分支 V_1, V_2 , 若回路 C 只在一个连通分支中 (譬如只在 V_1 中), 则 C 与 S 没有公共边; 若回路 C 在 $V_1 \cup V_2$ 中, 则从 V_1 中顶点到 V_2 中顶点, 从 V_2 中顶点到 V_1 中顶点, 因为是回路, 所以必偶数次经过 S 中的边, 故连通图中任一回路 C 与任一边割集 S 有偶数 (包括 0) 条公共边。

【例 9.3.6】 设 T 是二元正则树, i 为 T 中的分支点数, t 为 T 中的树叶数。证明: $i = t-1$

证明 由正则树定义可知, 二元正则树的每个分支点都有 2 个儿子, 所以 T 中有 $2i$ 条边。由于树中的顶点数比树叶数多 1, 故 T 中有 $2i+1$ 个顶点。已知 T 中有 i 个分支点和 t 片树叶, 因此 $2i+1 = i+t$, 即 $i = t-1$ 。

习题九

1. 画出所有非同构的 6 阶无向树。

2. 无向树 T 中有 7 片树叶, 3 个 3 度顶点, 其余都是 4 度顶点, T 中有多少 4 度顶点? 根据 T 的度数列, 画出 3 棵非同构的无向树。

3. 无向树 T 中有 n_2 个 2 度顶点, n_3 个 3 度顶点, $\cdots n_k$ 个 k 度顶点, T 中有多少个 1 度顶点?

4. 设有无向图 G 如图 9-1。

- (1) 画出 G 的关于完全图的补图 \bar{G} 。
- (2) 画出 \bar{G} 的所有不同构的生成树。

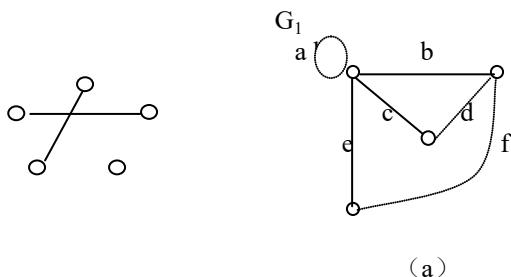


图 9-1

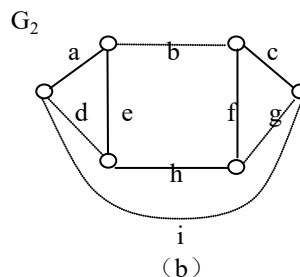


图 9-2

5. 如图 9-2 所示两个无向图, 其中实线边所示生成子图 T 为 G 的一棵生成树 T

- (1) 指出 T 的所有弦, 及 T 所对应的基本回路系统。
- (2) 指出 T 的所有树枝, 及 T 所对应的基本割集系统。

6. 求图 9-3 所示的 2 个带权图的最小生成树, 计算他们的权, 并写出关于这棵生成树的基本割集系统和基本回路系统。(请自己对图的边进行标定)

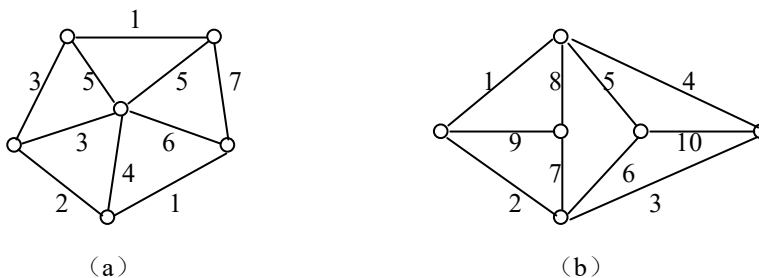


图 9-3

7. 证明: 对于 n 个顶点的数, 其顶点度数之和为 $2n-2$

8. 对于 $n \geq 2$, 设 d_1, d_2, \cdots, d_n 是 n 个正整数, 且 $\sum_{i=1}^n d_i = 2n-2$, 证明: 存在顶点度数为

d_1, d_2, \cdots, d_n 的一棵树。

9. 证明: $n > 2$ 的连通图 G 至少有两个顶点, 将它们删去后得到的图仍是连通图。

10. 设 T 是 $k+1$ 阶无向树, $k \geq 1$. G 是无向简单图, 已知 $\delta(G) \geq k$, 证明 G 中存在与 T 同构的子图。

11. 在连通图中, 对给定的一棵生成树, 设 $D = \{e_1, e_2, \cdots, e_k\}$ 是一个基本割集, 其中 e_1 是树枝, e_2, \cdots, e_k 是生成树的弦, 则 e_1 包含在对应于 e_i ($i = 2, \cdots, k$) 的基本回路中, 且 e_1 不包含在任何其它的基本回路中。(提示: 可利用例 9.3.5 的结论)

12. 在连通图中, 对给定的一棵生成树, 设 $C = \{e_1, e_2, \cdots, e_k\}$ 是一条基本回路, 其中 e_1 是弦, e_2, \cdots, e_k 是树枝, 则 e_1 包含在对应于 e_i ($i = 2, \cdots, k$) 的基本割集中, 且 e_1 不包含在任何其它的基本割集中。

13. 证明在二元正则树中, 边的总数等于 $2(t-1)$, 其中 t 是树叶的数目。

14. 证明二元正则树有奇数个顶点

15. 求出对应于图 9-4 所给树 (a) 和森林 (b) 的二元位置树。

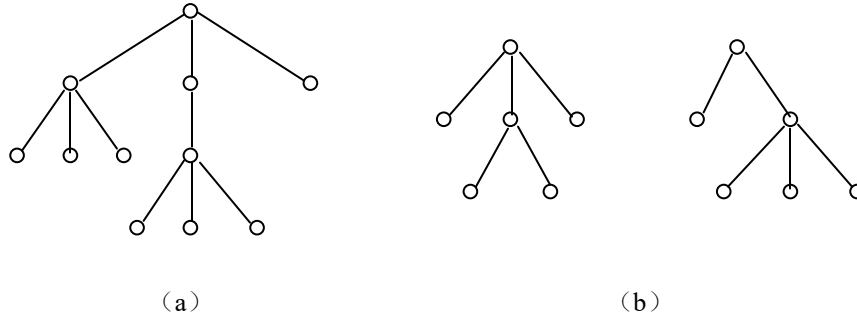


图 9-4

16. 画出公式 $(p \vee (\neg p \wedge q)) \wedge ((\neg p \rightarrow q) \wedge \neg r)$ 的根树表示。

17. 给定权 1, 4, 9, 16, 25, 36, 49, 64, 81, 100

(1) 构造一棵最优二元树。

(2) 构造一棵最优三元树。

(3) 说明如何构造一棵最优 t 元树。

18. 通讯中 a, b, c, d, e, f, g, h 出现的频率为

a: 25% b: 20% c: 15% d: 15% e: 10% f: 5% g: 5% h: 5%

通过画出相应的最优二元树, 求传输它们的最佳前缀码, 并计算传输 10000 个按上述比例出现的字母需要多少个二进制数码。

19. 构造一个与英文短语 “good morning” 对应的前缀码, 并画出该前缀码对应的二叉树, 再写出此短语的编码信息。