离散数学

Discrete Mathematics

吴梅红

厦门大学计算机科学系

E-mail: wmh@xmu.edu.cn



第八章 树 Tree

- 树是一种特殊图, 在CS中具有非常重要的应用。
- 本章所讲回路均指初级回路(圈)或简单回路。

8.1 无 向 树

定义8.1 连通无回路的无向图T称为无向树, 常用T表示树

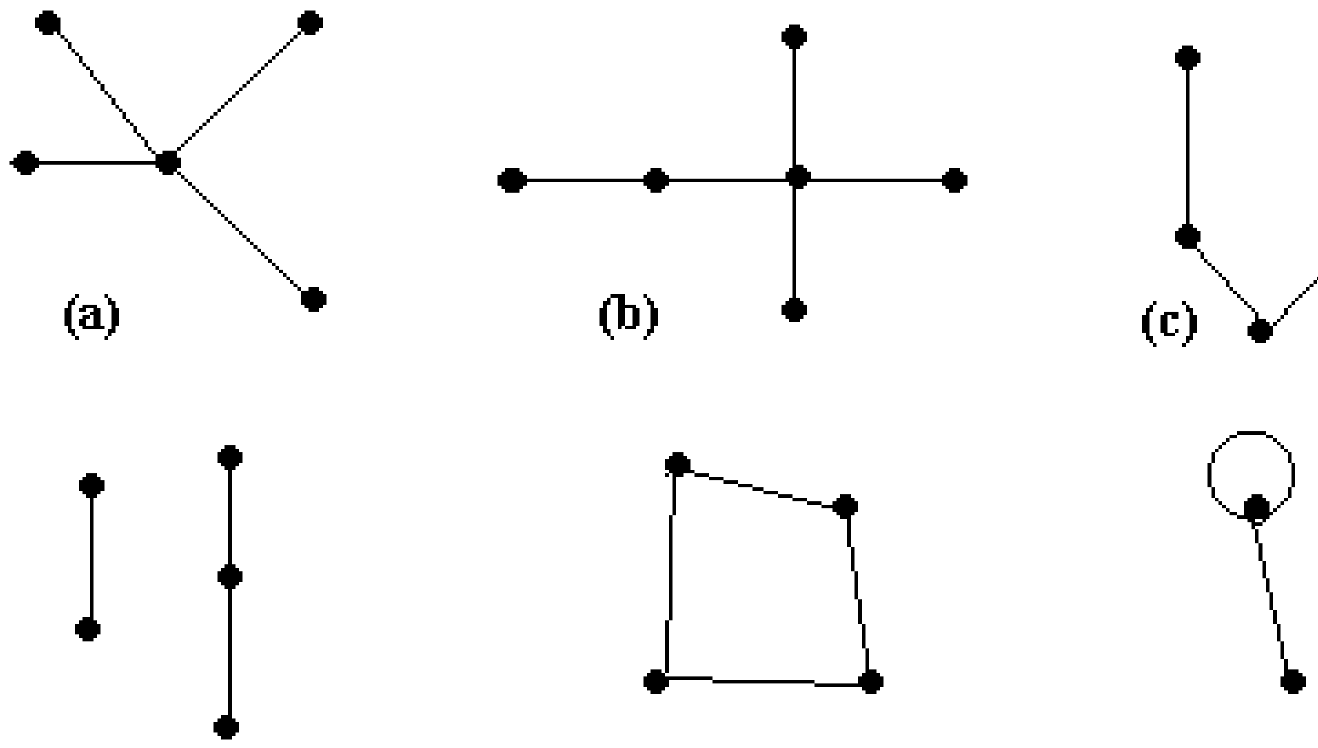
若无向图G至少有两个连通分支且每个连通分支都是树,

则称G为森林(Forest)。平凡图称为平凡树。 ■

- 设$T = <V, E>$为一棵无向树, $\forall v \in V$, 若 $d(v) = 1$, 则称

  v为T的树叶; 若$d(v) \geq 2$, 则称v为T的分支点。 ■

- 若T为平凡树, 则T既无树叶, 也无分支点。



(d) (e) (f)

例 图中的(a), (b)和(c)都是树; 而(d), (e)和(f)都不是树。只有(d)是森林。 /* 树是连通的森林

- 定理9.1给出树的特征性质, 它们是树的充分必要条件, 可看成树的等价定义。

定理 8.1 对一个(n, m)无向图G, 下列各命题等价。

1) G是树(连通, 无回路connected acyclic)。

2) G的任二顶点之间存在唯一的一条路径。

3) G中没有圈, 且$m = n – 1$。

4) G是连通的, 且$m = n – 1$。 /* 连通分界线

5) G是连通的, 且G中任何边均是桥。

/*树T中分支点必为割点。树是最弱的连通图

6) G中没有圈, 但在G中任二不同顶点 u, v 之间增加边 (u, v), 所得图含惟一的一个圈。 /*循环证明法

1) G是树 (连通, 无回路)。

2) G的任二顶点之间存在唯一的一条路径。

证 1) $\Rightarrow$ 2):

- 由G的连通性可知, $\forall$ u, v $\in$V, u, v之间存在通路,

  设P_1为u, v之间的一条通路, 若P_1不是路径, 则G中必有回路, 这与G中无回路矛盾。所以P_1为路径。

- 又若P_1不是u, v之间惟一的路径,

  设P_2是u, v之间不同于P_1的又一条路径, 则必存在边

  $e_1' = (v_x, v_1')$只在P_1上或只在P_2上。

- 不妨设e_1'在P_2上, 若还有与e_1'相邻的边e_2'在P_2上, 而不在P_1上, 得通路$e_1' e_2'$ (或$e_2' e_1'$)在P_2上, 不在P_1上,
- 继续这一过程, 最后得$e_k' = (v_k', v_y)$只在P_2上, $e_1'e_2'\ldots e_k'$记为P(x, y)为只在P_2上而不在P_1上的通路, 且v_x与v_y是P_1与P_2的公共顶点,
- 则P(x, y)并上P_1上v_x与v_y之间的一段路径得图G中一条回路, 这与G中无回路矛盾, 于是u, v之间的路径是唯一的。

2) G的任二顶点之间存在唯一的一条路径。

⇒ 3) G中没有圈, 且$m = n - 1$。

证 首先证明G中没有圈，若G中存在关联顶点v的自环, 则v到v存在两条路径，长度分别为0 (dist)和1 (自环), 这与已知条件惟一路径矛盾。

- 若G中存在长度大于等于2的圈, 则圈上任何二不同顶点之间均存在两条不同路径, 这与已知条件矛盾。
- 下面用强归纳法证明 $m = n - 1$。对n施行归纳:
- 当$n = 1$时, 因为G中无圈, 因而$m = 0$, 此时G为平凡图, 显然有$m = n - 1$, 故结论为真。

- 设对$n \le k$ $(k \ge 1)$ 时结论成立。
- 当$n = k + 1$, 此时G图至少有一条割边, 设$e = (u, v)$ 为G中一条割边, 则 $G - e$必有两个连通分支G_1, G_2

(否则$G - e$中u到v还有通路，因而G中含过u, v的回路, 导致G中含圈)。

- 设n_i , m_i 分别为G_i 中的顶点数和边数,

则 $n_i \le k$, $i = 1, 2$。

由强归纳假设知, $m_i = n_i - 1$ $(i = 1, 2)$。

- 于是, $m = m_1 + m_2 + 1$ /* (u, v)

$$= n_1 - 1 + n_2 - 1 + 1 = n - 1。$$

3) G中没有圈, 且$m = n - 1$。

$\Rightarrow$ 4) G是连通的, 且$m = n - 1$。

- 只要证明G是连通的。采用反证法。
- 否则设G有s ($s \geq 2$)个连通分支$G_1, G_2, \ldots, G_s$,

G_i 中均无圈, 因而G_i 均连通无回路, 即它们都是树。

由 1) $\Rightarrow$ 2) $\Rightarrow$ 3)可知, $m_i = n_i - 1$ (m_i, n_i 分别为G_i 的边数和顶点数, $i = 1, 2, \ldots, s$), 于是

$$m = \sum_{i=1}^{s} m_i = \sum_{i=1}^{s} n_i - s = n - s$$

由于$s \geq 2$, 这与已知条件$m = n - 1$矛盾。

4) G是连通的, 且 $m = n - 1$。

$\Rightarrow$ 5) G是连通的, 且G中任何边均是桥。

- 只要证明G中每条边均为桥。
- 任意的边 $e \in E(G)$,

  均有 $|E(G - e)| = n - 1 - 1 = n - 2$。

  由定理 ($m \geq n - 1$)可知,

  $G - e$不连通, 故e是桥。

5) G是连通的, 且G中任何边均是桥。

$\Rightarrow$ 6) G中没有圈, 但在G中任二不同顶点u, v之间增加边(u, v), 所得图含惟一的一个圈。

- 由于G中每条边均为桥, 因而G中不可能含圈,

  又因为G连通, 所以G为树。

- 由1) $\Rightarrow$ 2)可知, $\forall u, v \in V$, 且$u \neq v$,

  则 u, v之间存在惟一的一条路径P(u, v),

  则 $P(u, v) \cup (u, v)$为图$G \cup (u, v)$中惟一的一个圈。

6) G中没有圈, 但在G中任二不同顶点u, v之间增加边(u, v), 所得图含惟一的一个圈。

$\Rightarrow$ 1) G是树 (连通, 无回路)。

- 只要证明G是连通的。
- 由于$\forall$u, v$\in$V, u $\neq$ v, (u, v)$\cup$G产生惟一的圈C,

  则C – (u, v)为u到v的通路, 因而u, v连通。

  由u, v的任意性可知, G是连通的。 ■

推论 设G是由 r棵树构成的一 (n, m) 森林, 有关系式

$$m = n - r。$$

证 设G的第 i 个分支有n_i个顶点, m_i条边 ($1\le i\le r$),

则 $m = \sum_{i=1}^{r} m_i = \sum_{i=1}^{r}(n_i - 1) = \sum_{i=1}^{r} n_i - r = n - r$ 。

定理8.2设T是n阶非平凡的无向树, 则T至少有两片树叶.

证 设T有x片树叶, 由$m = n - 1$及握手定理可知

$$2n - 2 = 2m = \sum_{v_i \in V(T)} d(v_i) \geq x + 2(n - x) = 2n - x$$

$$\Rightarrow x \geq 2 \qquad \blacksquare$$

推论 阶大于2的树T必有割点。

证 由$n = m - 1$知道T至少含有一个度数≥ 2的分支点v,

再由定理9.1 (T是连通的, 且T中任何边均是桥)。

$T - \{v\}$不是连通的, 所以v是割点。 $\blacksquare$

推论 树中分支点必为割点。 $\blacksquare$ 　　树中边均为桥。

树是连通性最弱($n = m - 1$)的图; 但最经济的连通图。

例 8.1 已知一棵无向树T中4度, 3度, 2度的分支点各一个, 其余的顶点均为树叶, 问T中有几片树叶?

解 设T中有几片树叶。则T的阶数 $n = 1+1+1 + x = 3 + x$, 由 $m = n - 1$, $m = 2 + x$。 由握手定理有 总度数

$4 + 2x = 2m = 4 + 3 + 2 + x = 9 + x$, 解得 $x = 5$片树叶。

例 8.2 满足例8.1中度数列(4, 3, 2, 1, 1, 1, 1, 1)的无向树在同构的意义下是惟一吗。

解 在同构意义下不惟一。

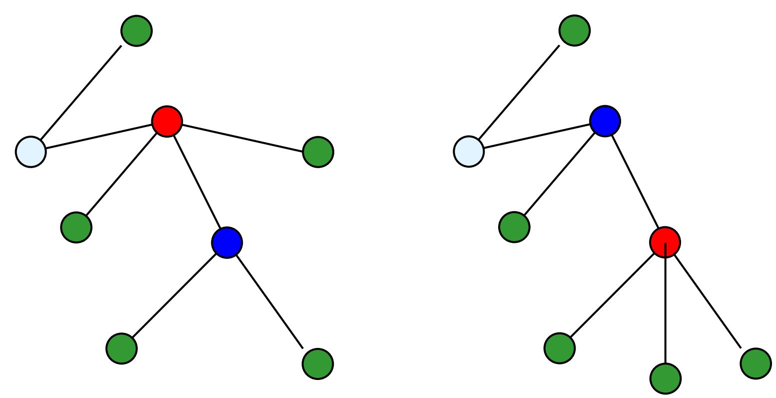


图 8.2

生成树 Spanning Tree

定义8.2 设T是无向图G的子图且为树, 则称T为G的树。

若T是G的生成子图并且为树, 则称T为G的生成树。

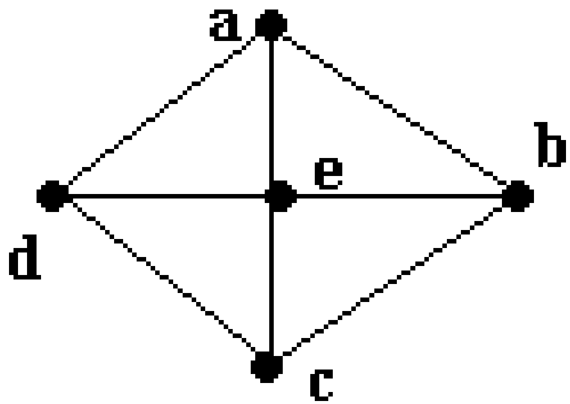
对任意的边$e \in E(G)$, 若 $e \in E(T)$, 则称e为T的树枝,

否则称e为生成树T的弦, 称所有弦的集合导出子图

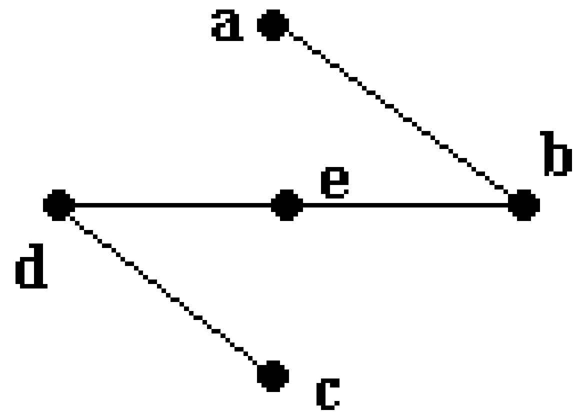
$G[E(G) - E(T)]$为生成树T的余树(或补), 记作T。 ∎

- 在以上定义中, 注意$\overline{T}$不一定是树。
- 一般来说, 连通图的生成树不是唯一的。
- 一个图G与它的生成树的差别在于G可能包含有回路, 而生成树不包含有回路。

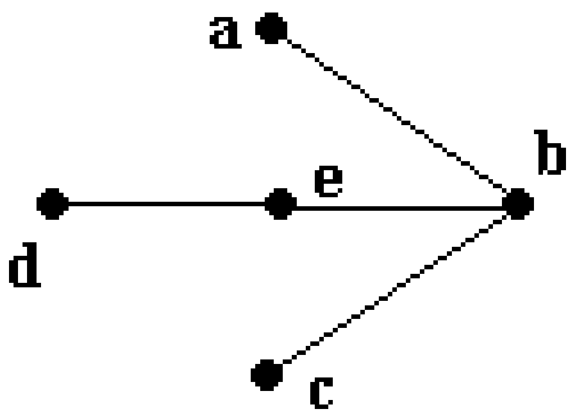
例 下图中只给出G的三棵不同的生成树。How many …



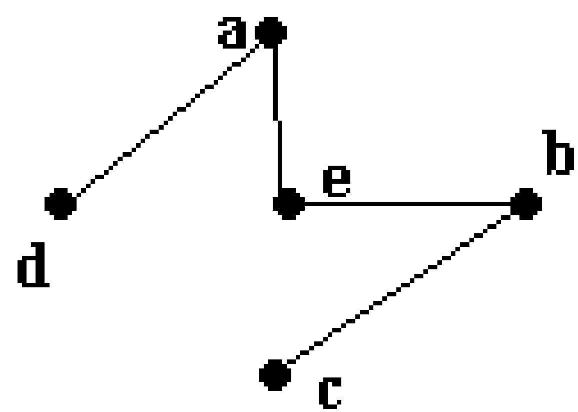
(G)



(T1)



(T2)



(T3)

$$5^3 = 125$$

定理 8.3 无向图G具有生成树 $\Leftrightarrow$ G是连通的。

证 必然性 $\Rightarrow$ G由生成树连通。 充分性 $\Leftarrow$ (破圈法)

- 若G中无圈, 则G为树, 当然G为自己的生成树。
- 若G中含圈, 任取一个圈C, 随便删除C上任何一条边, 所得图仍然是连通的(顶点不变)。
- 继续这一过程, 直到最后得到的图无圈为止。
- 设最后的图为T, 则T是连通的无圈且是G的生成子图, 所以T是G的生成树。 ∎

推论1 无向简单连通图G的生成树不一定是惟一的;

证 (1) 非同构, 存在不同的边就认为是不同的

(2) 在同构的意义下, 可以存在非同构的生成树。

推论2 设G为n阶无向简单连通图, 则 $m \geq n-1$。

证 G是n阶连通图, G必含有生成树T, $m \geq |E(T)|$

T中 $|E(T)| = n-1$, 所以$m \geq n-1$ 。

推论3 设T是n阶m条边的无向连通图G的一棵生成树,

则T的余树$\overline{T}$中含$m-n+1$条边。

证 若图的生成树T的树枝大于$n-1$, 则必有回路在其中.

- 若生成树T的树枝小于$n-1$, 则必不连通。
- 这两种情形不可能, 所以生成树T的树枝为$n-1$, 弦便有$m-(n-1) = m-n+1$条。 ∎

推论4 设T是连通图G中一棵生成树, $\overline{T}$为T的余树, C为G中任意一圈, 则$E(\overline{T}) \cap E(C) \neq \varnothing$。/*C至少一条边在$\overline{T}$

定理 设T是无向连通图G中的一棵生成树, e为T的任意一条弦, 则T∪e中含G的只含一条弦其余边均为树枝的圈(基本回路), 而且不同的弦对应的圈是不同。

证 设 $e = (u, v)$, 由定理8.1(6)可知,

u, v之间存在惟一的路径P(u, v)在T中,

则P(u, v)∪e中只含弦e其余边均为树枝的圈。

- 显然当 e_1, e_2 为不同的弦时,

  e_2 不在 e_1 对应的圈 C_{e_1} 中,

  e_1 不在 e_2 对应的圈 C_{e_2} 中。 ■

定义8.3 设T是n阶m条边的无向连通图G的一棵生成树，

设$e_1, e_2, \ldots, e_{m-n+1}$为T的弦，设$C_r$是T添加$e_r$产生的G中

只含 e_r 其余边均为树枝的圈，称C_r为G对应T的弦 e_r

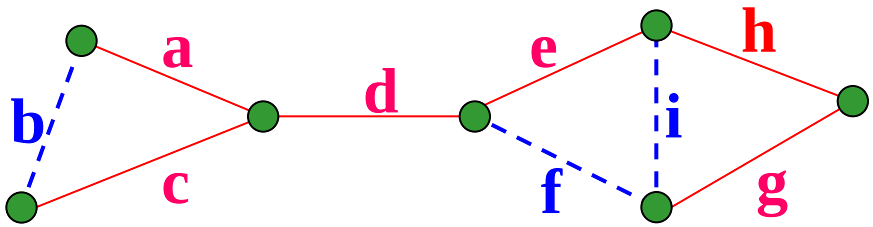
的基本回路或基本圈，$r = 1, 2, \ldots, m - n + 1$,

并称$\{C_1, C_2, \ldots, C_{m-n+1}\}$为G对应T的基本回路系统，

称 $m - n + 1$ 为G的 圈秩，记作 $\xi(G)$。 $\blacksquare$

- $G(n, m)$ 图的任一生成树，恒有$m - n + 1$条弦，

  从而有 $\xi(G) = m - n + 1$ 个基本回路.



例 设图G如图 所示，求G的基本回路系统。

解 在G中红实线边所示的子图为G的一棵生成树。

弦b, f, i 对应的基本回路分别为

C_b = bca, C_f = fghe, C_i = igh,

$C_{基} = \{C_b, C_f, C_i\}$,

- 定义8.3反映了 生成树的弦与基本回路的关系。
- 由定义不难看出, 无向连通图(n, m) 的不同生成树对应的基本回路系统可能不同, 但基本回路系统中的元素圈个数均为G的圈秩 $\xi(G) = m - n + 1$ 。
- 圈秩是为了弄破G的所有圈而必须从G中删去的边的最小数目, 即 任一生成树的总弦数。

定理 设T是连通图G的一棵生成树, e为T的一条树枝,

则G中存在只含树枝e, 其余元素均为弦的边割集。

设e_1, e_2是T的不同的树枝,

则它们对应的只含一条树枝的边割集是不同的。

证 由定理8.1知道, e为T的桥, 因而T – e有两个连通分支,

设为T_1和T_2。令G的边割集 S_e =

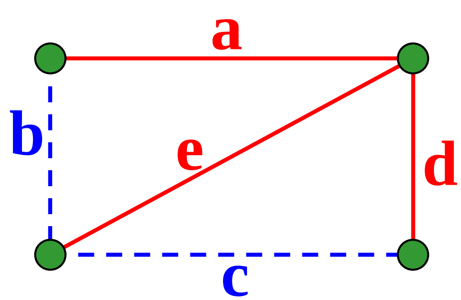
$\{e \mid e \in E(G)$且 e的两个端点分别属于$V(T_1)$和$V(T_2)\}$

- 显然$e \in S_e$, 且除e外S_e中元素全是弦, 由构造可知,

若e_1, e_2是不同的树枝, 它们对应的割集 S_{e1}与S_{e2}不同。

定义 8.4 设T是n阶连通图G的一棵生成树, $e'_1, e'_2, \ldots,$ e'_{n-1}为T的树枝, S_l是G的只含树枝e'_l的割集, 则称 S_l为G的对应生成树T 由树枝 e'_l 产生的基本割集, $l = 1, 2, \ldots, n-1$, 并称$\{S_1, S_2, \ldots, S_{n-1}\}$为G对应T的基本割集系统, 称$n-1$为G的割集秩, 记作$\eta(G)$。 ■

- $\eta(G)$即任一生成树的总树枝数。
- 对每一树枝e, $T - e$ 分为两个连通分支T_1, T_2, 那么e及两端点分别在T_1, T_2中的弦组成一基本割集。



例 无向图G如图 所示, 求G 的基本割集系统。

解 图中实线边所示的图为G的一棵生成树。

a, e, d为T的树枝, 对应的基本割集分别为

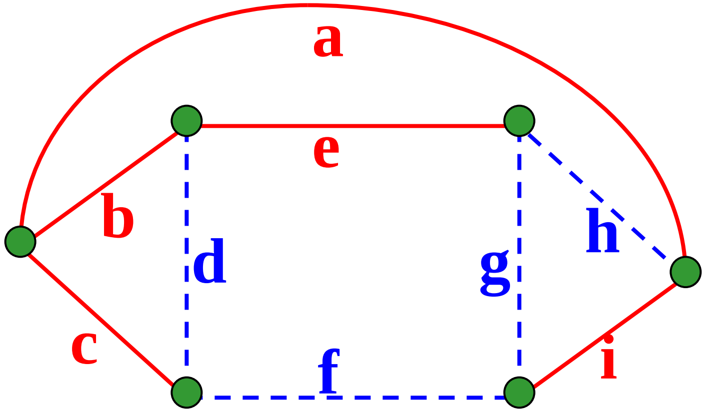
$S_a = \{a, b\}$, $S_d = \{d, c\}$, $S_e = \{e, b, c\}$,

$S_{基} = \{S_a, S_d, S_e\}$。

- 定义8.4反映了 生成树的树枝与基本割集的关系。
- 由定义不难看出, 连通图的

  不同生成树对应的基本割集系统可能不同,
- 但基本割集系统中的元素个数均为 $\eta(G) = n - 1$。
- (n, m) 图G的任一生成树恒有 $n - 1$ 条树枝,

  从而有 $n - 1$ 个基本割集。



例8.5左图中**红线**边所示子图为G的一棵**生成树**T, 求G对应T的**基本回路系统**和**基本割集系统**

解 $n = 6$, $m = 9$, T有 $m - n + 1 = 4$ 条弦 d, f, g, h, 因而有

4个基本回路: $C_d = d\ c\ b$, $C_f = f\ c\ a\ i$, $C_g = g\ e\ b\ a\ i$, $C_h = h\ e\ b\ a$, 基本回路系统为 $\{C_d, C_f, C_g, C_h\}$。

- T有 $n - 1 = 5$ 条树枝 $c, b, a, , e, i$, 因而有5个基本割集:

$S_c = \{c, d, f\}$, $S_b = \{b, d, g, h\}$, $S_a = \{a, h, g, f\}$, $S_e = \{e, g, h\}$, $S_i = \{i, g, f\}$。

基本割集系统为 $\{S_c, S_b, S_a, S_e, S_i\}$。 ■

- 若G本身是树, 则G只含有一棵生成树。
- 若G为非树的连通标定图, 则G所含的生成树多于一棵
- 我们感兴趣的有两个问题:

1. 如何寻找连通标定图G的所有生成树及其计数公式。

- 在电网络的拓扑分析中，需要找出所有可能不同的生成树。而全部不同的生成树是一个很大的数目。
- 在G(n, m)标定完全图中, 不同的生成树有n^{n-2}个之多。
- 当n增大时，寻找图G的所有生成树是非常困难的, 涉及到计算复杂性问题。

2. 寻找最小生成树。　　　　　　/*第14章, DS

- 想架设一个通讯线路, 把若干城市连接起来, 怎样才能使所用的线路总长最短(或时间最少, 资源最少)呢?

  问题的实质就是求带权的最小生成树问题。

- 下面讨论标定(允许同构)的无向图中生成树的个数。

- 设$G = \langle V, E \rangle$为无向连通图, $V = \{v_1, v_2, \dots, v_n\}$,

  即G是顶点标定顺序的标定图,

- 设T_1, T_2是G的两棵生成树, 若 $E(T_1) \neq E(T_2)$,

  则认为T_1与T_2是G的不同的生成树。

定义 8.5 给定图$G = <V, E>$, 设 $W: E \to R$ (R为实数集),

任意的边$e = (v_i, v_j)$ (G为有向图时, $e = <v_i, v_j>$),

设 $W(e) = w_{ij}$, 称w_{ij}为边e上的权, 并将w_{ij} 标注在边e上,

称G为带权图, 此时常将G记为$<V, E, W>$。

- 设$G_1 \subseteq G$, 称 $\sum\limits_{e \in E(G_1)} W(e)$为子图$G_1$的权, 记为$W(G_1)$。

定义8.6 设无向连通带权图$G = <V, E, W>$,

G中带权最小的生成树称为G的最小生成树。 ▮

- 若$G = <V, E, W>$中，V为n个城市的集合，

  E是城市之间道路的集合，而对于任意的边$e = (v_i, v_j)$，

  $W(e) = w_{ij}$ $(w_{ij} > 0)$为造公路e的造价，则G中每棵最小生成树都是n个城市间的总造价最小的公路网。

- 求最小生成树的 避圈法(Kruakal算法)
- 设$G = <V, E, W>$为n阶m条边的简单无向连通图,

  E中边按它们所带权的大小排序编号,

  即 $W(e_1) \le W(e_2) \le \ldots \le W(e_m)$。
- 用避圈法求G中最小生成树的算法如下:
- 开始 令 $T_0 = <V, \varnothing>$, $i \leftarrow 1$, $j \leftarrow 0$。 /*j为T下标

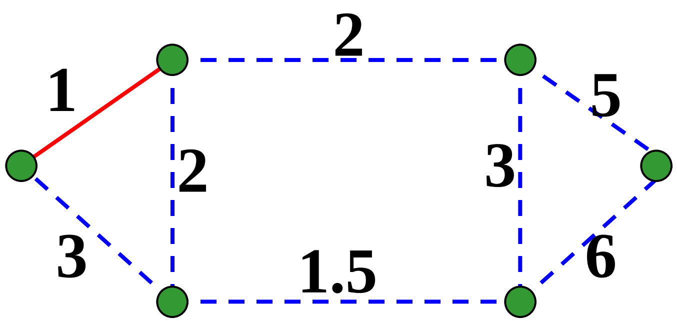
  1. 若$T_j \cup \{e_i\}$含圈转2, 否则转3。 /*i为e下标

  2. $i \leftarrow i+1$, 转1。 /*选最小边且避圈

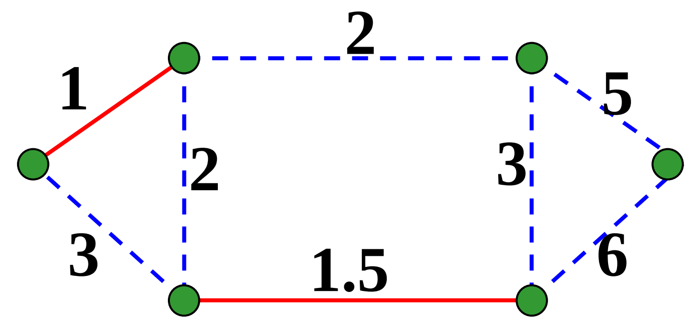
  3. 令$T_{j+1} = T_j \cup \{e_i\}$, $j \leftarrow j+1$。

  4. 若$j = n - 1$, 停止, 否则转2。

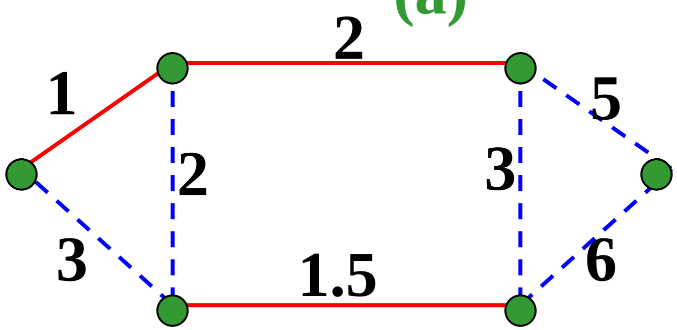
- 由于G是连通图, 当算法结束时, 得到的G的**n–1**阶无圈子图T_{n-1}, 由定理**8.1**可知T_{n-1}是G的生成树。
- 又被算法留在T_{n-1}外的边均为T_{n-1}的弦，由算法可知, 这些弦在它们所对应的基本回路中是带权最大的边, 可知, T_{n-1}是G的最小生成树。
- 综上所述, **Kruakal**的避圈法是正确的。



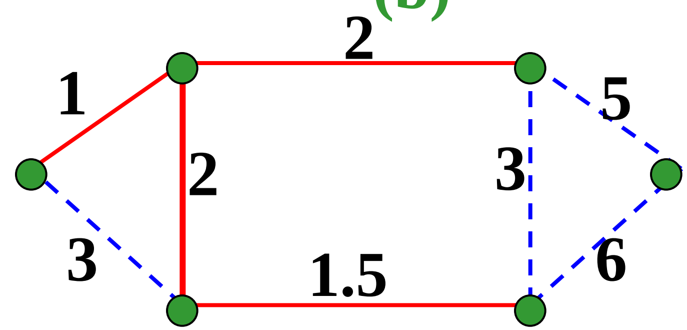
(a)



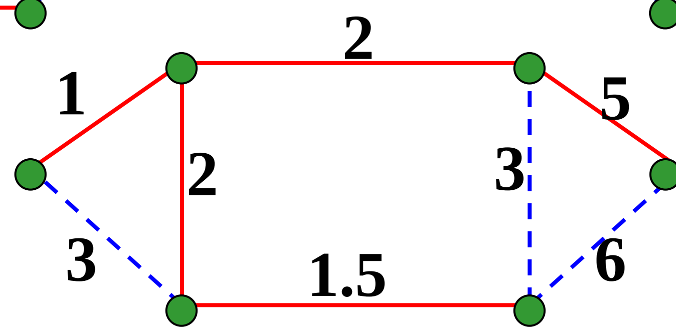
(b)



(c)



(d)



(e)

例14.6 用**避圈法**求图14.11所示图的最小生成树。

解　图14.12给出了算法的计算过程，**实线边**表示**树枝**。所得生成树T为**最小生成树**，W(T) = 11.5。

2. 破圈法

- 由Rosenstiehl于1967年给出。
- 设$G = \langle V, E, W \rangle$为n阶m条边的无向连通图,
- 用破圈法求G中最小生成树的算法如下:
- 开始 令 $G_0 = G$, $k \leftarrow 0$。

1. 若G_k中不含圈, 转2。否则C为G_k中一个圈,

   e_k为C上带权最大的边, 令$G_{k+1} = G_k - e_k$;

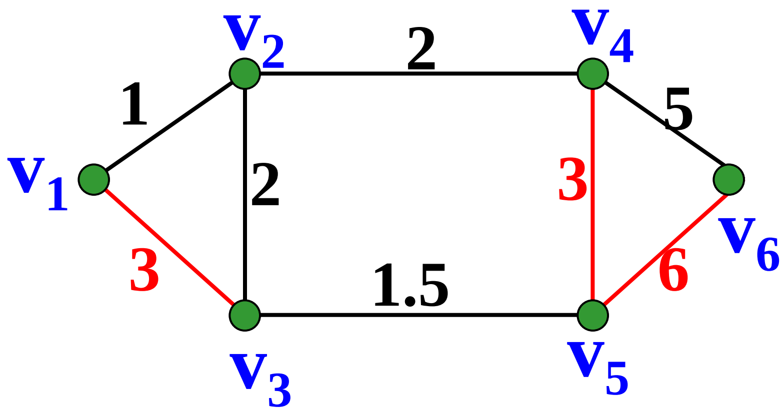
   $k \leftarrow k+1$, 重复1。

2. 结束。

- 结束时, G_k为G中最小生成树。
- 当G中圈较少时, 用破圈法比避圈法好些。

例 用破圈法求图中的

最小生成树



解 先选哪个圈都没关系。

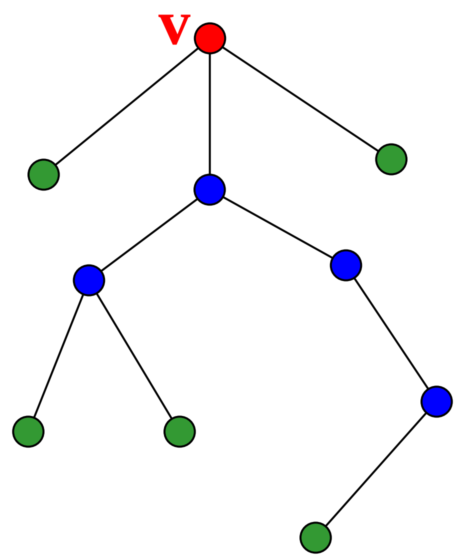
- 若先选的圈为$v_4v_5v_6v_4$, 则删除边(v_5 , v_6),
- 再选圈$v_1v_2v_3v_1$, 删除边(v_1 , v_3),
- 再选圈$v_2v_4v_5v_3v_2$, 删除边(v_4 , v_5),
- 最后得生成树T, W(T) = 11.5。
- 虽然以上2种方法得到的最小生成树均是同一棵, 但这无一般性, 一般说来, 图中的最小生成树不一定惟一。

§8.2 根 树 Root Tree

- 若有向图D的基图是无向树, 则称D为有向树。在有向树中, 最重要的是根树。在DS, DB等专业课中是VIP。

定义 8.7 若有向图树T是平凡树 或 T中有一个顶点的入度为0, 其余顶点的入度均为1, 则称T为根树。

- 入度为0的顶点称为树根, 入度为1出度为0的顶点称为树叶, 入度为1出度不为0的顶点称为内点。内点和树根统称为分支点。
- 从树根到T的任一顶点v的路径长度 称为v的层数*l* (v), 树根v_0为第0层, 层数最高的顶点的层数称为树高。



- 在根树中, 由于各有向边的方向的一致性, 因而画根树时, 省掉有向边的箭头, 并将树根放在最上方。
- 左图给出一棵高为4的根树, v为它的树根, 它有5片树叶, 5个分支点, 其中有4个内点。

定义 设T为一棵根树可看成一棵家族树, v_i, $v_j \in V(T)$,

若v_i可达v_j, 则称v_i是v_j的祖先, v_j是v_i的后代;

若v_i邻接到v_j, 则称v_i是v_j的父亲, v_j是v_i的儿子;

若v_j, v_k的父亲相同, 则称v_j, v_k是兄弟。 ▌ /*

定义 8.8 如果将根树T的每一层上的顶点都规定次序，则称T为有序树。标出的次序不一定是连续的数。 ■

定义 8.9 设T为一棵根树，

(1)若T的每个分支点至多有r个儿子，则称T为r叉(元)树；

(2)若T的每个分支点都恰有r个儿子，称T为r元正则树；

(3) 若T是r元正则树，且每个树叶的层数均为树高，则称T为r元完全正则树；

(4) 若T是r树且为有序树，则称T为r元有序树；

(5) 若T是r元完全正则树 且是有序的，则称T为r元完全正则有序树。 ■

- 2元有序树和2元正则有序树在DS中居重要地位。

最 优 树

- 设T是m元树, 若对T的**每片树叶指定一个实数**, 则称T为**带权的m元树**。

定义8.10 设二元树T有t片树叶, 权分别为w_1, w_2, …, w_t, 称$W(T) = \sum_{i=1}^{t} w_i \, l(v_i)$为T的权, 其中$l(v_i)$为$v_i$的层数。在所有带权为$w_1$, w_2, …,w_t 的t片树叶的二元树中, 其权最小的二元树称为**最优二元树**, 简称**最优树**。

- 下面介绍最优树的**Huffman算法**。
- 给定实数w_1,w_2,…,w_t, 且$w_1 \le w_2 \le \ldots \le w_t$, 算法如下

(1) 连接以w_1, w_2为权的两片树叶,

得到分支点带权为$w_1 + w_2$;

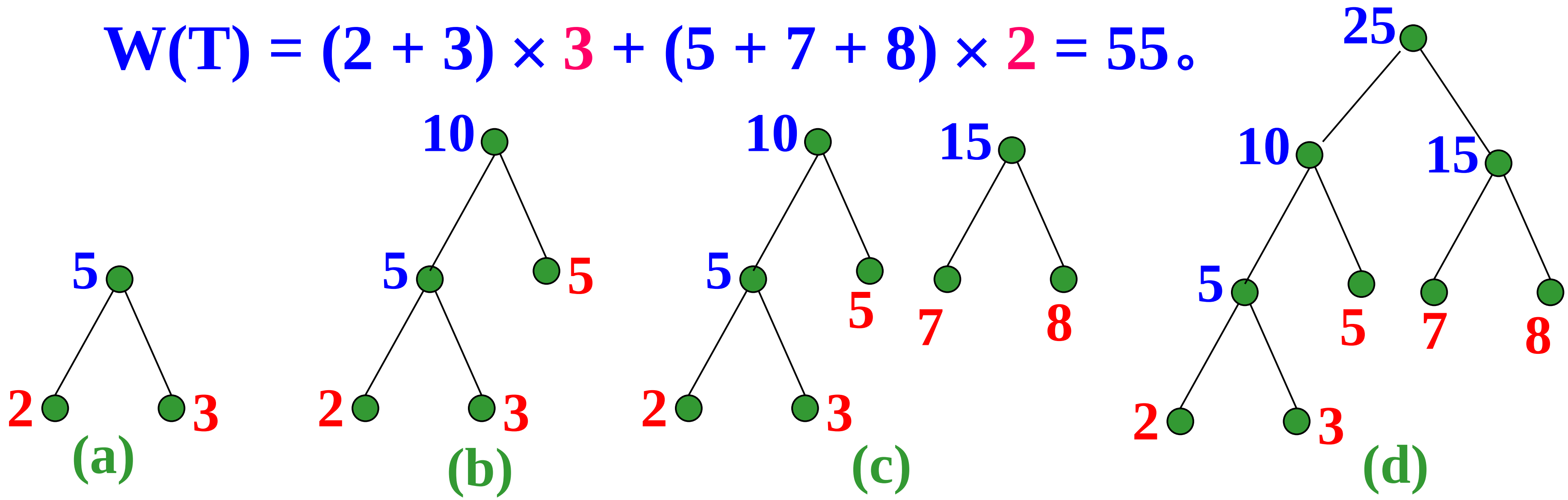
(2) 在$w_1 + w_2$, w_3, …, w_t中选两个最小的权, 连接它们对应的顶点(未必是树叶)又得到新的分支点及所带的权;

(3) 重复2, 直到形成$t - 1$个分支点, t片树叶为止。

例 用Huffman算法求带权最为2, 3, 5, 7, 8的最优二元树.

解 求最优树的过程如下:

$$W(T) = (2 + 3) \times 3 + (5 + 7 + 8) \times 2 = 55。$$



- 最优二元树用途之一是求最佳前缀码。
- 在通信工作中, 常用二进制数字0, 1组成的符号串(简称为二元码)来表示数字、字母、汉字等,
  用长为n的二元码最多可表示2^n个符号。ASCII码。
- 若传输的符号出现的频率相同, 用等长的表示法很好。
- 若传输的符号出现的频率不同, 用等长的码子传输它们就造成浪费, 因而想办法利用不等长的码子来传输。
- 英文单词中, 字母q, x, z等出现的频率较低。

定义 8.11 设$\beta = \alpha_1\alpha_2\ldots\alpha_n$为长为n的符号串, 称其子串$\alpha_1$, $\alpha_1\alpha_2$, …, $\alpha_1\alpha_2\ldots\alpha_{n-1}$分别为长为1, 2, …, n–1的前缀。

设$A = \{\beta_1, \beta_2, \ldots, \beta_m\}$为一个符号串集合,

若对于任意的β_i, $\beta_j \in A$, $i \neq j$,

β_i, β_j 互不为前缀, 则称A为前缀码。

若β_i (i= 1,2,…,m)中只出现0与1, 则称A为二元前缀码。

- {0, 10, 110, 1111}, {1, 01, 001, 000}等均为前缀码, 而{1, 01, 101, 001, 0011}不是前缀码。
- 用二元树可以产生前缀码。

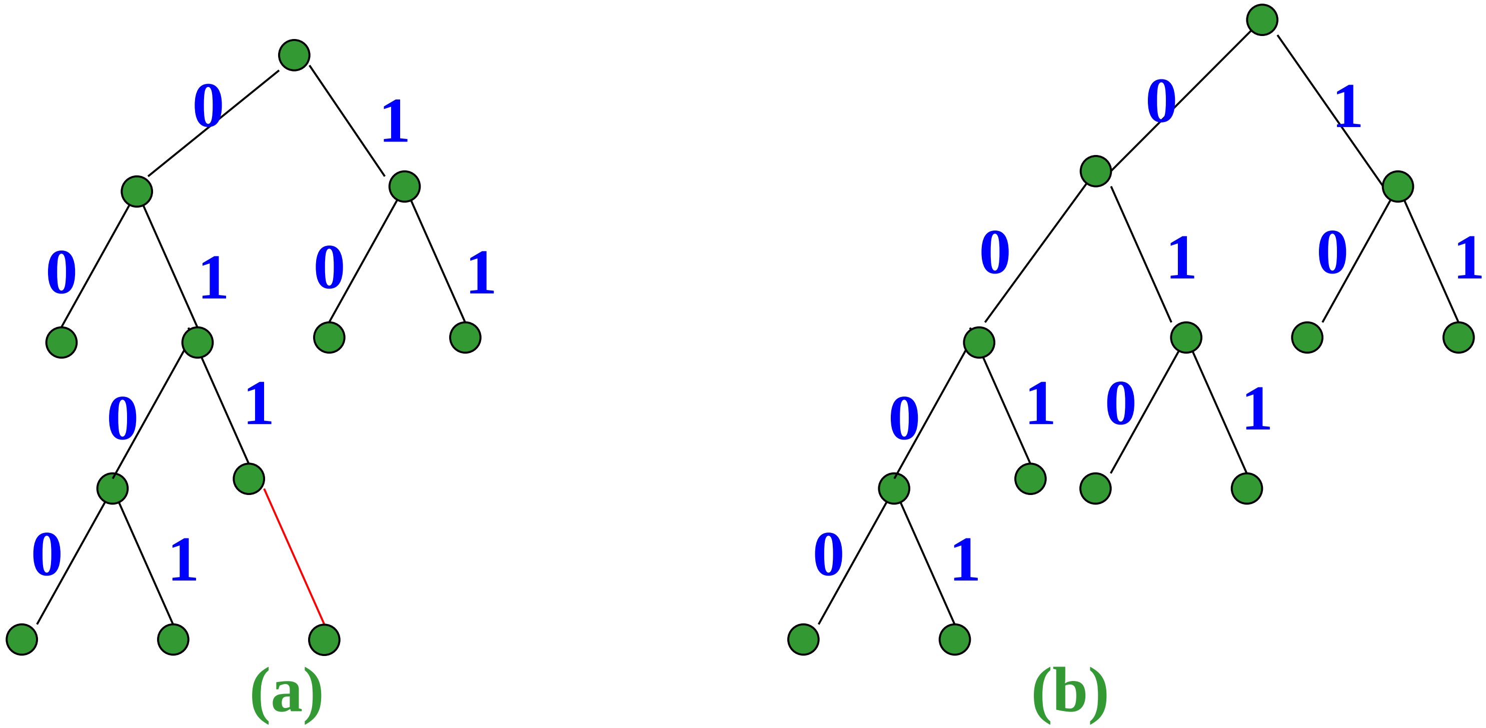
定理 8.4 一棵给定的二元树可以产生一个二元前缀码。

证 给定一棵二元树T, 设T有t片树叶。

对于T的任意分支点v, 若v有两个儿子,

在v引出的左边标上0, v引出的右边标上1。 /*左0右1

- 若v只有一个儿子, 将引出的惟一的一条边标上0或1。
- 从树根v_0到每片树叶的通路上所标的数字组成一个二元的符号串记在该片树叶处, 于是得到t个符号串β_1, β_2, …, β_t, 记A = {β_1, β_2, …, β_t}, 则B为前缀码。
- 这是因为第i片树叶v_i处的符号串β_1的前缀均在从树根v_0到v_i的通路上, 所以任意β_i, $\beta_j \in$A, β_i, β_j 互不为前缀, 即A = {β_1, β_2, …, β_t}为前缀码。



推论 一棵二元正则树可以产生惟一的一个前缀码。

图(a)所示二元树(非正则)产生的前缀码为{00, 0100, 0101, 0111, 10, 11}。图(b)所示二元正则树产生的前缀码为{0000, 0001, 001, 010, 011, 10, 11}。

- 当知道了要传输的符号的频率时, 可以用各符号出现的频率乘100当权, 用Huffman算法求最优二元树T, 所得的前缀码$\{\beta_1, \beta_2, \ldots, \beta_t\}$为最佳前缀码, 用这样的前缀码传输对应的符号可以使传输的二进制位数最省。

例8.7 在通信中, 八进制数字出现的频率为

0: 30%	1: 20%	2: 15%	3: 10%
4: 10%	5: 5%	6: 5%	7: 5%

求传输它们的最佳前缀码, 并讨论传输10^n($n\geq 2$)个按所给频率出现八进制数字比“3位等长传输法”提高

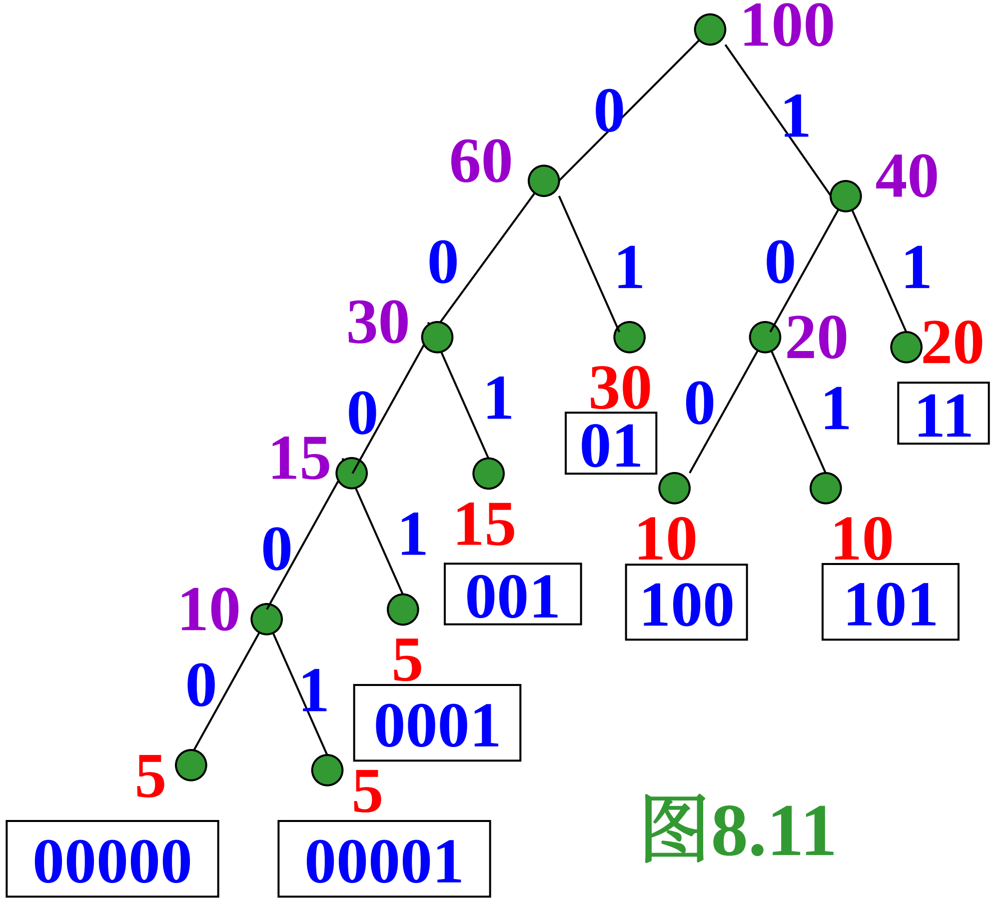


图8.11

效率百分之几?

解 $w_i = 100p_i$, $i = 1, 2, .., 7$, 按从小到大顺序为

$5 \le 5 \le 5 \le 10 \le 10 \le 15 \le 20 \le 30$,

所对应的最优树为图8.11所示.

- 八进制数字对应的前缀码为

0——01	1——11
2——001	3——100
4——101	5——0001
6——00001	7——00000

- **W(T)=275**，说明传输**100**个八进数字用**275**个二进制数字, 所以传输10^n个用$275\times10^{n-2}$个二进制数字。
- 而用“等长的码子”传10^n个八进制数字用3×10^n个二进制数字, 所以提高效率为

$$\frac{3\times10^n-2.75\times10^n}{3\times10^n}\approx 8\%$$

- 应该指出, 所求的最优树可能不只一棵, 但它们的权是相等的.

定义 设T为一棵根树, v∈V(T), 称v及其后代的导出子图 T_v 为T的以v为根的根子树。

- 2元正则有序树的每个分支点的两个儿子 导出的根子树分别称为该分支点的左子树和右子树。
- 对一棵根树的每个顶点都访问且仅访问一次称为行遍或周游一棵根树。
- 设T为一棵2元正则有序树, 按对树根、左子树、右子树

的不同的访问顺序主要有以下3种行遍方法: /*先左后右

(1) 中序行遍法 访问次序: 左子树, 树根, 右子树;

(2) 前序行遍法 访问次序: 树根, 左子树, 右子树;

(3) 后序行遍法 访问次序: 左子树, 右子树, 树根

- 红色表示为根子树的根。

(1) 中序行遍法

(d b (h e i)) a (f c g);

(2) 前序行遍法

a (b d (e h i)) (c f g);

(3) 后序行遍法

(d (h i e) b) (f g c) a。

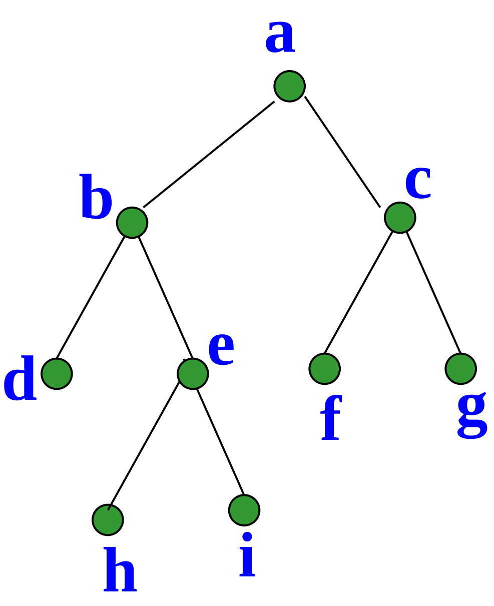


图 8.12

- 四则运算表达式可以存储在2元正则有序树上:
- 参加运算的数放在树叶上,

  并规定被减数、被除数放在左子树上,

  减数、除数放在右子树上,
- 分支点上放相应的运算符号,
- 从某两片树叶开始 按从低到高运算层次的顺序开始存放, 树根应该放的是最高层次的运算符。
- 然后根据不同的行遍方法访问根树T, 可以得到四则运算的不同的表达方法, 从而得到不同算法:

(1) 按中序行遍法访问T, 可以还原算式,

其特点是运算符夹在两个参加运算的数之间,

故称所得算式的表示法为中缀符号法。

(2) 按前序行遍法访问T, 在所得表达式中规定,

每个运算符对它后面紧邻的两个数进行运算,

并可以省掉表达式中的全部括号，称此种表达算式的

方法为前缀符号法, 或称波兰符号法。 /*函数名

(3) 按后序行遍法访问T, 在所得表达式中规定,

每个运算符对它前面紧邻的两个数进行运算,

仍可以省去全部括号，称此种表达算式的方法为后缀

符号法, 或称逆波兰符号法。

例 9.7 设有算式 中缀表达

$((a\times(b+c))\times d-e)\div(f+g)\div(h\times(i+j))$

(1) 将以上算式存入一棵2元正则有序树中;

(2) 分别写出上式的波兰符号法和逆波兰符号法表达的形式。

解 (1) 树T如图9.12所示。

(2) 波兰符号法表达的形式为

$\div\div-\times\times a+b\,c\,d\,e+f\,g\times h+i\,j$

逆波兰符号法表达的形式为

$a\,b\,c+\times d\times e-f\,g+\div h\,i\,j+\times\div$

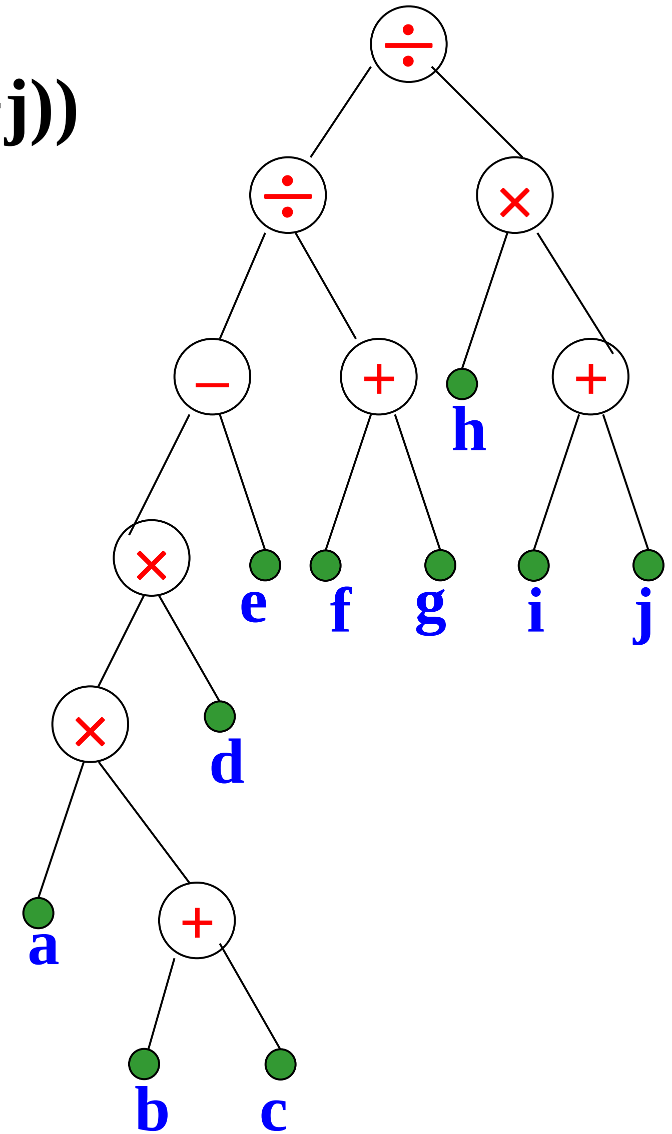


图9.12