定理7.6.1 树定义的等价条件

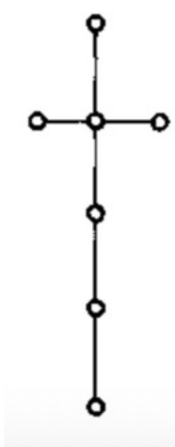
设 $G = \langle V, E, \Psi \rangle$ 是 n 阶无向图，则以下条件等价:

i) G 是连通的和非循环的。(树的定义)

ii) G 无自圈，且当 $v, v' \in V$时，皆有唯一的一条从 v 至 v′ 的基本路径。

iii) G 是连通的，且当 $v, v' \in V$时，$e \notin E$, $\Psi' = \{<e, \{v, v'\}>\}$ 时,, $G+\{e\}_{\Psi'}$ 有唯一的一条回路。

iv) G 是连通的，且当$e \in E$时， $G - e$ 是非连通的。



v) G是连通的 且$n(E) = n - 1$。

vi) G是非循环的且有$n(E) = n - 1$。

定理 7.6.2 阶大于 1 的树至少有两个端点。

森林

树是非循环的连通无向图，如果去掉对连通性的要求，就得到森林的概念。

定义7.6.2 每个分支都是树的无向图称为森林。

定理 7.6.3 如果森林 F 有 n 个结点，m 条边和 k 个分支，则 $m=n-k$。

证明：n 个顶点的树有 n-1 条边，设每个分支有 n_i 个顶点，则：$n_1+n_2+\ldots+n_k = n$。

森林一共有 $(n_1-1)+(n_2-1)+\ldots+(n_k-1) = n-k = m$ 条边

因此 $m=n-k$。

生成树(Spanning Tree)、生成森林

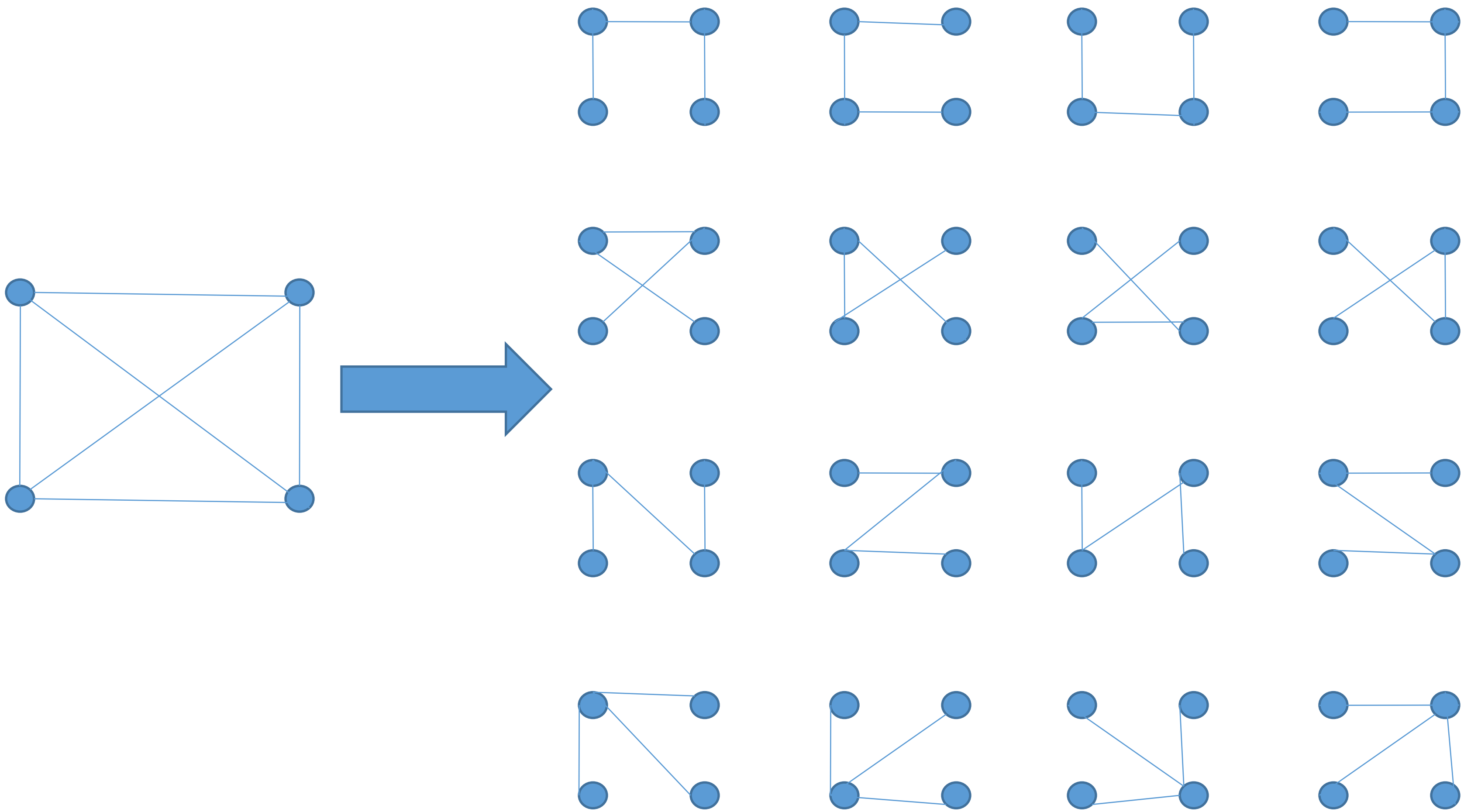
定义7.6.3

- 如果树T是无向图G的生成子图，则称T 为G的生成树。
- 如果森林F是无向图G的生成子图，则称F为G的生成森林。

定理 7.6.4

- 每个无向图都有生成森林。
- 无向图G有生成树当且仅当G是连通的。

生成树(Spanning Tree)示例



连通图的生成树构造方法

1、避圈法：

添加 e_1，…，e_i，在添加的每一步均保证：e_{i+1}不与 $\{e_1$，…，$e_i\}$ 的任何子集构成回路。

2、破圈法：

在 G_0 (即G)中去掉 e_1 得到G_1，

在 G_1中去掉 e_2 得到G_2，

在G_2中去掉 e_3得到G_3，…

其中 e_i 为 G_{i-1} 中某条回路中的边，直到没有回路，即把G中的所有回路均挑破！！！

定理：设无向图G连通，则G至少有一个生成树。

该定理的证明过程实际上是求生成树的算法：

输入：连通无向图G

输出：生成树 T_G

（1）$i \leftarrow 1$，$G_0 \leftarrow G$;

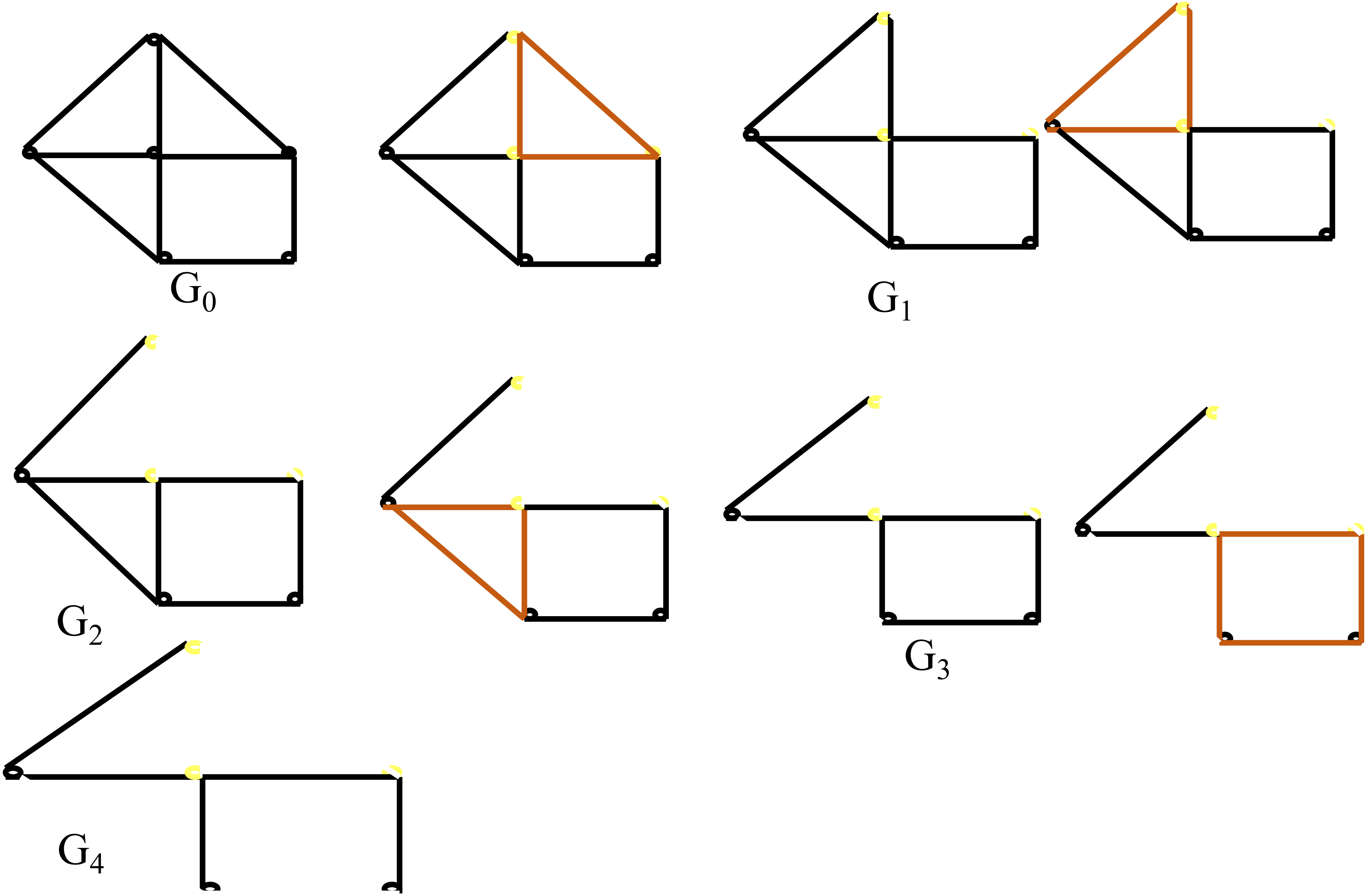
（2）若G_i无圈，则$T_G \leftarrow Gi$并终止；否则转（3）；

（3）找出G_i中任何一圈α_i，并从α_i中去掉任何一边x_i，$G_{i+1} \leftarrow G_i - x_i$;

（4）$i \leftarrow i+1$，转（2）

“破圈法”：逐次破掉图G中所有的圈，并保证每破一圈时都得到G的一个连通生成子图，因而最后得到的TG保证是G的生成树。

用破圈法求图G的生成树



最小生成树-Minimum Spanning Tree (MST)

定义7.6.4 (i)设〈G，W〉是加权图，G′⊆G。G′中所有边的加权长度之和 称为 G′的加权长度。

(ii) 设 G 是连通无向图，〈G，W〉是加权图，G的所有生成树中加权长度最小者称为〈G, W〉的最小生成树。

贪心法求解最小生成树常用的有两种算法:

(1) Prim's MST algorithm (prim算法).

(2) Kruskal's MST algorithm(kruskal算法).

Prim算法是基于点的，而Kruskal算法是基于边的。

最小生成树求法（避圈法、破圈法）

按 “ 避圈法 ” 求最小生成树：

设 G 是有m条边的n阶连通无向图，

1° 把 G 的m条边按加权长度递增的顺序排成 $e_1, e_2, \ldots, e_m$;

2° $T \leftarrow \varnothing$;

3° $j \leftarrow 1$，$i \leftarrow 1$;

（i记录正在扫描的边的下标；j记录T中边数是否已达n-1）

4° 若 $j = n$ 则算法结束。

5° 若 G 的以 $T\cup\{e_i\}$ 为边集合的子图没有回路，

则 $T \leftarrow T\cup\{e_i\}$ 且 $j \leftarrow j+1$;

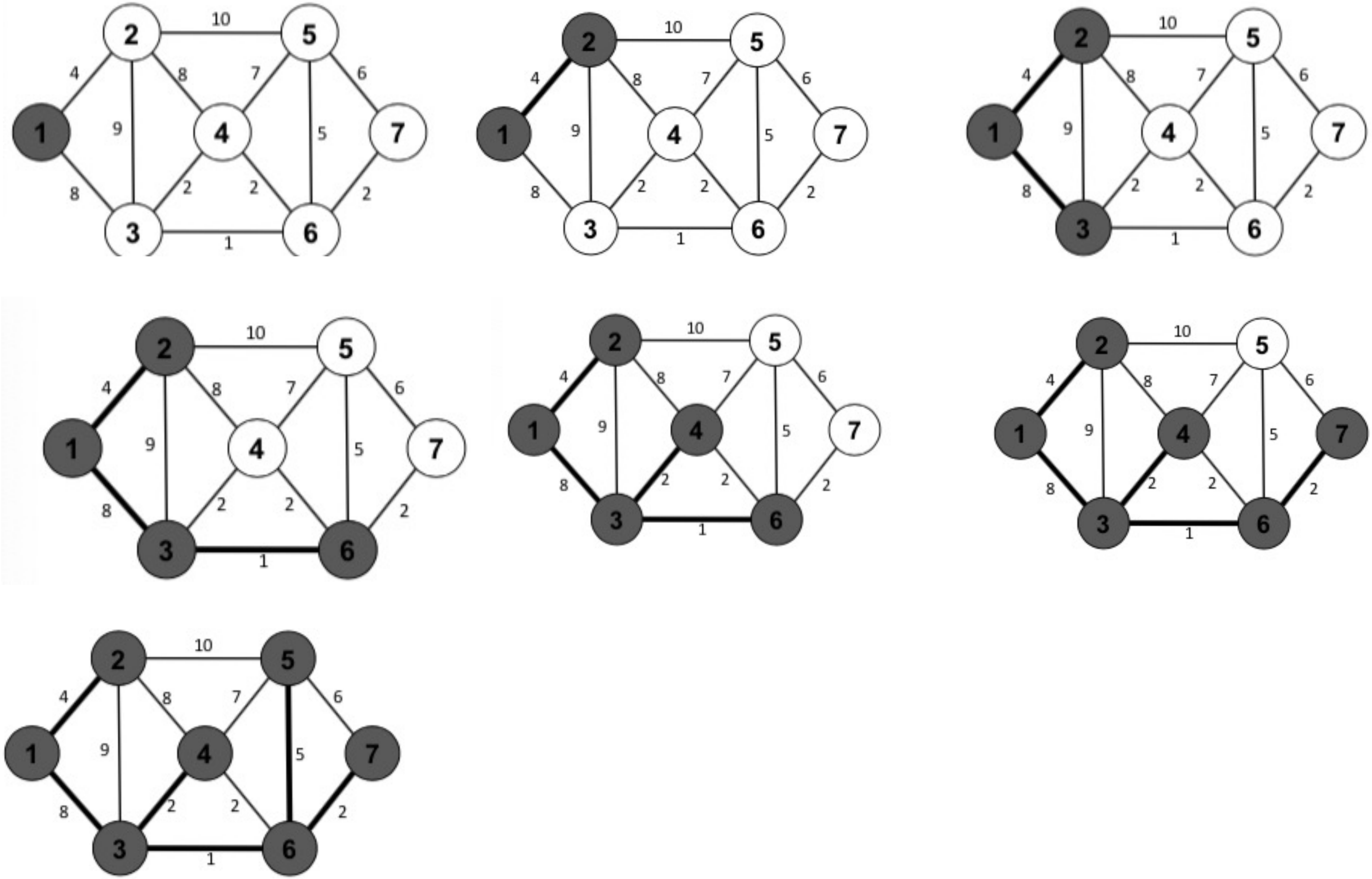
6° $i \leftarrow i+1$，转向 4°

算法结束时，T 即为所求的最小生成树的边集。

最小生树算法--Prim算法

- 用于连通无向图，贪心算法
  1. 维护Tree结构
  2. 初始E={} V={v} //任取节点v
  3. 循环n-1次
     - 选择一条边(v1, v2), 满足
       - $v1 \in V$, $v2 \notin V$,
       - (v1, v2)权值最小
     - $E = E \cup (v1, v2)$
     - $V = V \cup \{v2\}$

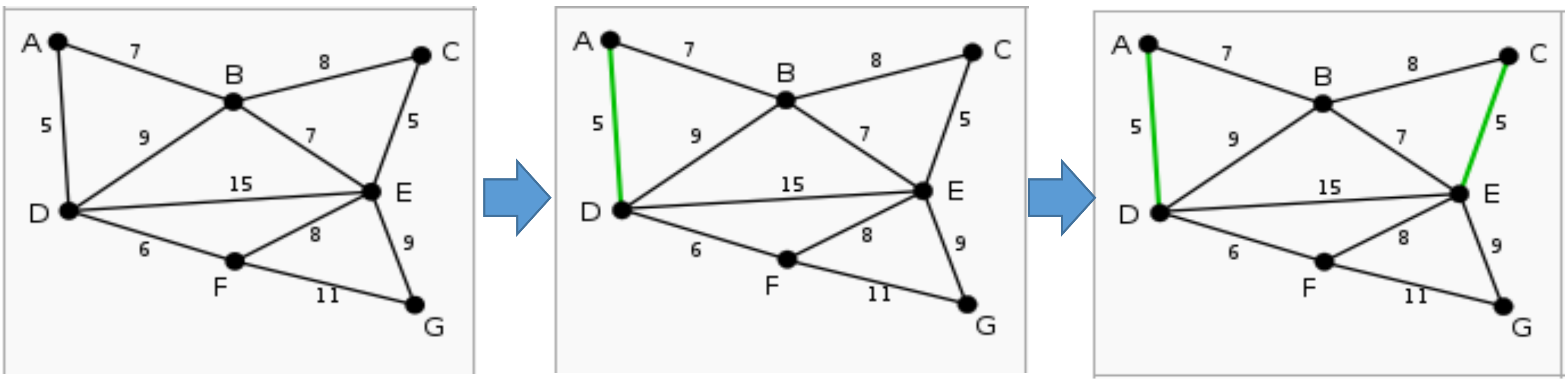
Prim算法：示例



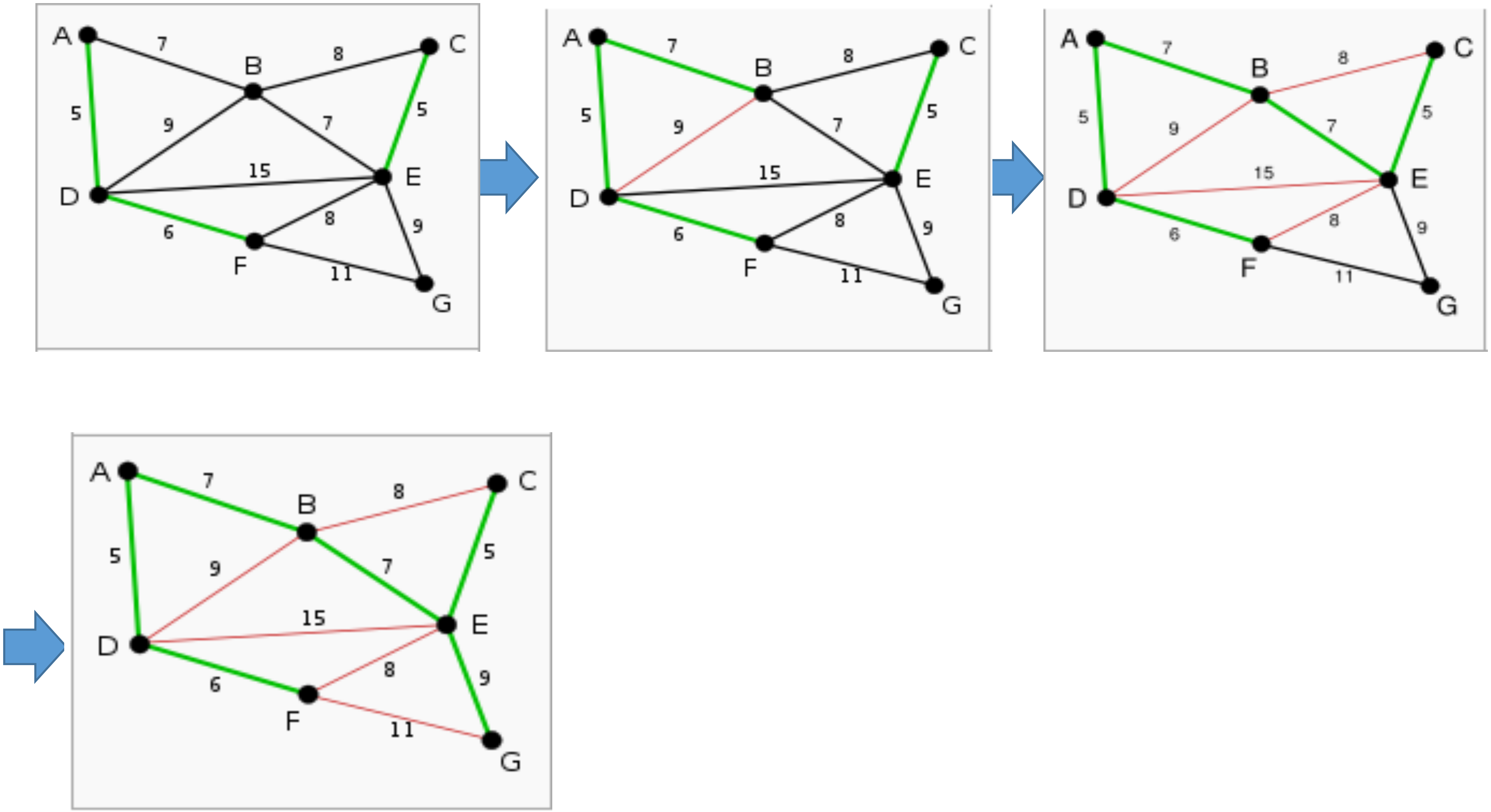
Kruskal算法

- 贪心算法
  - ➢将边按从小到大排序
  - ➢按顺序选择每条边，只要与已选择边不构成圈，就选择。
  - ➢终止条件
    - 已经选择了**n-1**条边；
    - 如果处理所有边，仍然不够**n-1**条，则说明图不连通

Kruskal's Algorithm - Example



Kruskal's Algorithm - Example



枝、弦

定义 7.6.5 设 T 是连通无向图 G 的生成树，称 T 的边为枝，而 G 的不属于 T 的边称为 弦。

问题：连通图 G 的边 e 是枝还是弦？

- 与给定的生成树 T 密切相关。
- 对于 G 的某个生成树 T， e 是枝，而对于 G 的另一个生成树 T_1， e 却可能是弦。
- 但是，对于 G 的任何生成树，枝的数目和 弦的数目都是固定的。

定理7.6.5 设 G 是有 m 条边的 n 阶连通无向图，则对于 G 的任何生成树 T，都有 $n-1$ 个枝和 $m-n+1$ 个弦。

圈秩、余圈秩

定义 7.6.6 若 n 阶无向图 G 有 m 条边和 k 个分支，则 G 的余圈秩 $r = n - k$，圈秩 $\mu = m - n + k$。

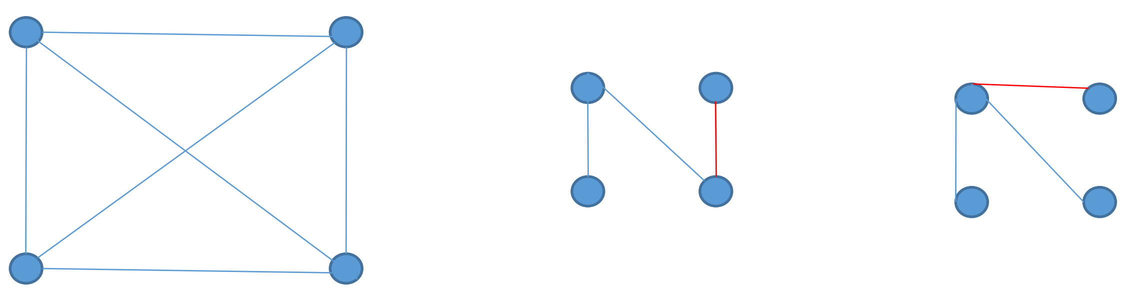
- 如果 G 是连通图（$k = 1$），则 G 的余圈秩 r 是枝的数目， 圈秩 μ是弦的数目。

基本回路（圈）

- 由定理7.6.1知，如果在生成树中增加一条弦，则恰产生一个回路。

定义7.6.7 （基本回路）设 T 是连通无向图 G 的生成树，G 的只包含一条弦的回路称为基本回路。

- 基本回路的概念与生成树相关联
  - 某回路对这个生成树是基本回路，而对另一个生成树却未必是基本回路。



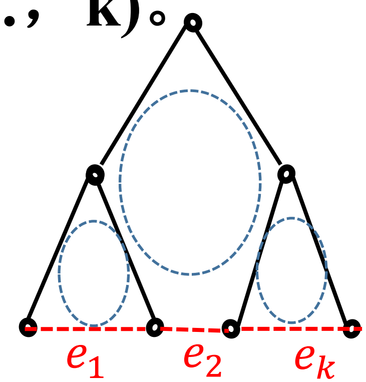
定理 7.6.6 设 T 是连通无向图 G 的任意生成树。

i） 基本回路的数目等于 G 的圈秩μ;

ii) 对于G的任意回路 C，总可以找到若干个基本回路 C_1，C_2，…，C_k，使 C 与 $C_1 \oplus C_2 \oplus \ldots \oplus C_k$ 的差别仅在于孤立点。

证明：i) 显然。

ii) 设 C 是 G 的任意回路且C 包含 k 条弦，显然 k>0，设这 k 条弦是 e_1，e_2，…，e_k，C_i 是包含 e_i 的基本回路 (i =1，2，…，k)。



令 $C' = C_1 \oplus C_2 \oplus \ldots \oplus C_k$，则 C' 包含的弦也是 e_1，e_2，…，e_k。

因此，$C \oplus C'$ 中的边都是枝，则 $C \oplus C'$ 是非循环的 。

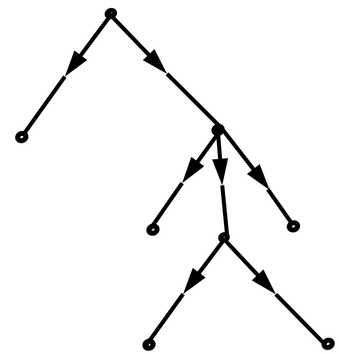
下面证明 $C \oplus C'$ 是零图。

若 $C \oplus C'$ 不是零图，必有一分支是阶大于 1 的树，根据定理7.6.2，$C \oplus C'$ 有端点。

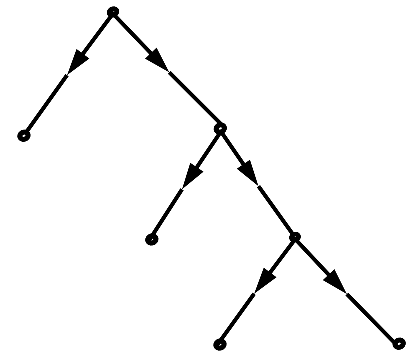
另，因为 C 和 C' 都是欧拉图，所以$C \oplus C'$ 是欧拉图。这与 $C \oplus C'$ 有端点矛盾，故 $C \oplus C'$ 必为零图，即C 与 C'的差别仅在于孤立点。

有向树

定义7.6.8 一个结点的入度为 0，其余结点的入度均为 1 的弱连通有向图 称为有向树。其中，



(a)



(b)

i) 入度为 0 的结点称为根，

ii) 出度为 0 的结点称为叶，

iii) 出度大于 0 的结点称为分支结点，

iv) 从根至任意结点的距离称为该结点的级，

v) 所有结点的级的最大值称为有向树的高度。

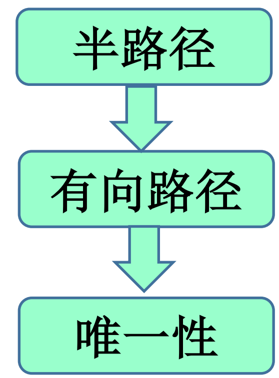
定理7.6.7 设 v_0 是有向图 D 的结点。 D 是以 v_0 为根的有向树当且仅当从 v_0 至 D 的任意结点恰有一条路径。

证明: (必要性) 设 D ＝〈V, E, Ψ〉是有向树，v_0 是D的根。

因为 D是弱连通的，任取 v′∈V，则存在从 v_0 至 v′的半路径 P，

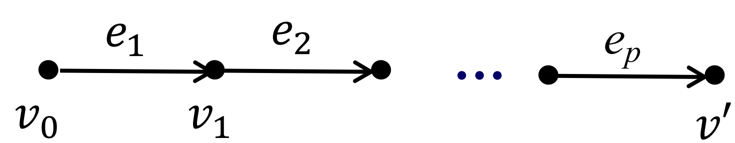
设 P为 $v_0\, e_1\, v_1 \ldots v_{p-1}\, e_p\, v_p$，其中 $v_p = v'$。

因为 $d_D^-(v_0) = 0$，所以 e_1 是正向边，因为$d_D^-(v_1) = 1$，所以 e_2 也是 正向边。



由归纳法可以证明：每个 e_i（$1 \le i \le p$）均是正向边。

故 P 为有向路径。

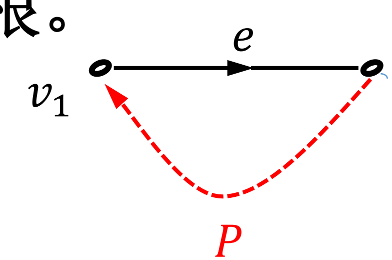


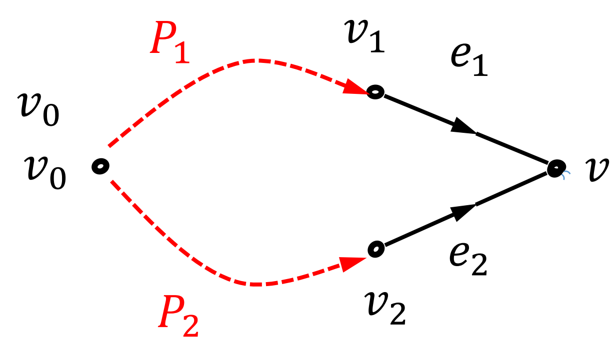
若从v_0至v′有两条路径 P_1 和 P_2，则P_1和P_2至少有一个公共点 的入度大于1，与 D 是有向树矛盾。

定理7.6.7 设 v_0 是有向图 D 的结点。D 是以 v_0 为根的有向树当且仅当从 v_0 至 D 的任意结点恰有一条路径。

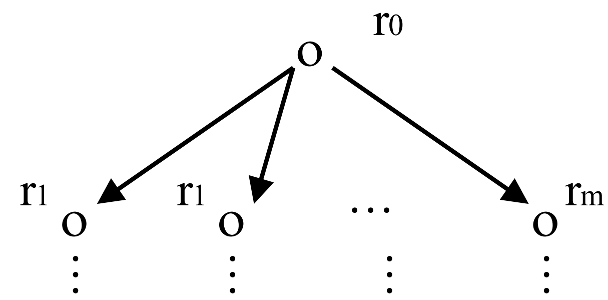
证明: (充分性) 若 $d_D^-(v_0) > 0$，则存在边e以v_0为终点（D弱连通），设v_1是e的起点，P是从v_0至v_1的路径，则在D中存在两条不同的从 v_0 至 v_0 的 路径 Pv_1ev_0 和 Pv_1ev_0 Pv_1ev_0，与已知条件矛盾，所以 $d_D^-(v_0) = 0$。

若$d_D^-(v) > 1$, 其中 v 是D的结点，则存在两条边e_1和e_2以v为终点。设e_1和e_2的起点分别是v_1和v_2，从v_0至v_1和从v_0至v_2的路径分别是P_1和P_2，则P_1e_1v和P_2e_2v是两条不同的从v_0至v的路径，与已知条件矛盾。所以，D是有向树，且v_0是D的根。





有向树的归纳定义



定义7.6.9 有向树归纳定义如下：

i) 平凡图是有向树，其结点称为该有向数的根。

ii) 设 m∈ I_+ , $D_1, D_2, \ldots, D_m$分别是以$r_1, r_2, \ldots, r_m$为根的有向树，并且两两不 相交，r_0不是 $\bigcup_{i=1}^{m} D_i$ 的结点，$e_1, e_2, \ldots, e_m$不是$\bigcup_{i=1}^{m} D_i$中的边，并且

$$\Psi: \{e_1, e_2, \ldots, e_m\} \rightarrow \{r_0, r_1, \ldots, r_m\}^2$$

定义为 $\Psi(e_i) = \langle r_0, r_i \rangle$ (i=1, 2, …, m)。

若 $G=\langle\{r_0, r_1, \ldots, r_m\}, \{e_1, e_2, \ldots, e_m\}, \Psi\rangle$，则

$D=G\cup \bigcup_{i=1}^{m} D_i$是有向树，$r_0$是D的根，并且称$D_1,D_2, \ldots, D_m$是D的子树。

有向森林

定义7.6.10 每个弱分支都是有向树的有向图，称为有向森林。

m元有向树

定义7.6.11 设 m∈N，D 为有向树。

i)如果D的所有结点出度的最大值为m，则称D为m元有向树。

ii)如果对于m元有向树 D 的每个结点v，皆有$d_D^+(v) = m$ 或 $d_D^+(v) = 0$，则称D为完全m元有向树。

- 完全二元有向树也称二叉树。

用途：字母和符号识别程序　　{+， -， *， / }

00　01　10　11

统计字母出现的频繁程度

两个问题？

- 编码问题
  - ➢假设ABCD四个字母：如何编码？
  - ➢出现频率是0.5,0.3,0.05,0.15，如何编码？

主题1：叶加权二叉树，Huffman编码

- ***树的存储计算***
  - ***二叉树具有特点和良好性质；***
  - ***对于不同类型树，是否能统一存储和计算？***

主题2：森林-树-二叉树的转化

叶加权二叉树

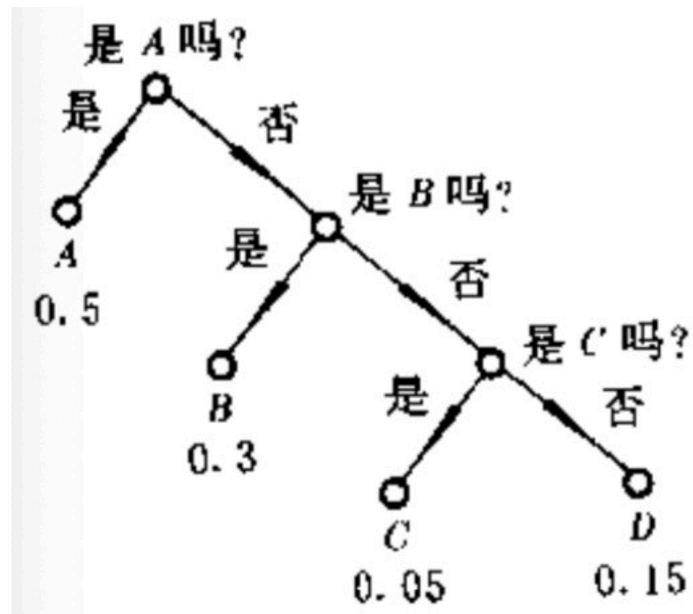
定义7.6.12 (i) 设 V 是二叉树 D 的叶的集合， W： $V \rightarrow R_{+}$，则称〈D， W〉为叶加权二叉树。

(ii) 对于 D 的任意叶 v，称 W(v) 为 v 的权，

称 $\Sigma_{v \in V}$ (W (v)·L(v)) 称为〈D，W〉的叶加权路径长度，其中 L(v)为v的级。

- 用叶表示字母或符号，
- 用分支结点表示判断，
- 用权表示字母或符号出现的概率，

则叶加权路径长度就表示算法的平均执行时间。



右图的叶加权路径长度为：0.5 ·1+o.3 ·2+0.05 ·3+0.15 ·3 =1.7

最优二叉树

定义7.6.13 设〈D，W〉是叶加权二叉树。

如果对任一叶加权二叉树〈D′，W′〉，只要对于任意正实数r，D 和 D′中权等于 r 的叶的数目相同，就有〈D，W〉的叶加权路径长度不大于〈D′，W′〉的叶加权路径长度，则称〈D，W〉为最优的。

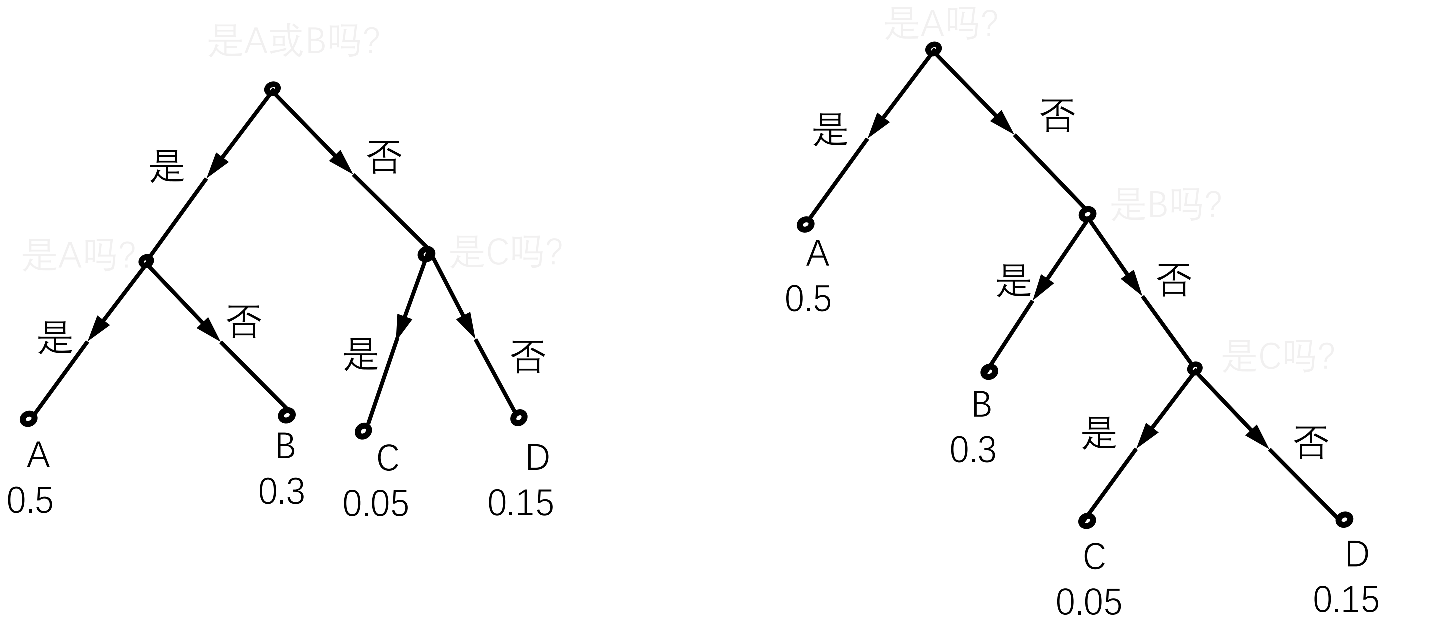


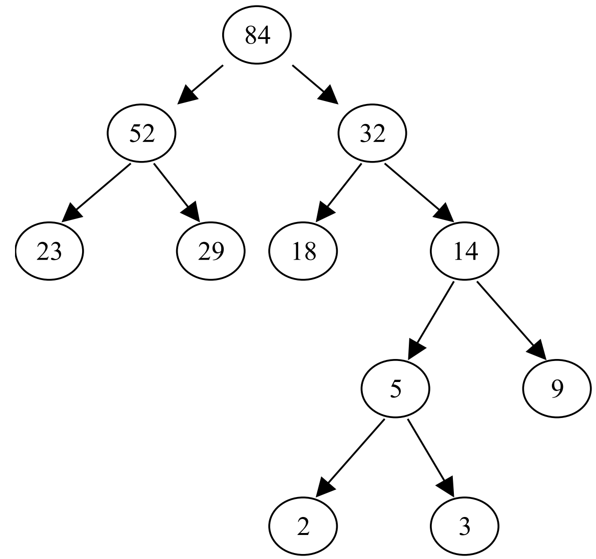
图7.6.5 用叶加权二叉树研究算法

最优二叉树求取

* 我们把求某问题的**最佳算法**归结为求**最优二叉树**。□

最优二叉树求取算法：（举例说明）

2	3	9	18	23	29
	5	9	18	23	29
		14	18	23	29
			32	23	29
			32		52
					84



- 将求 n 个叶的最优二叉树归结为求 n-1 个叶的最优二叉树。
- 所有**分支结点中的数值之和**就是**叶加权路径长度**

有序树、有序森林

定义7.6.14

(i) 为每一级上的结点规定了次序的有向树称为有序树。

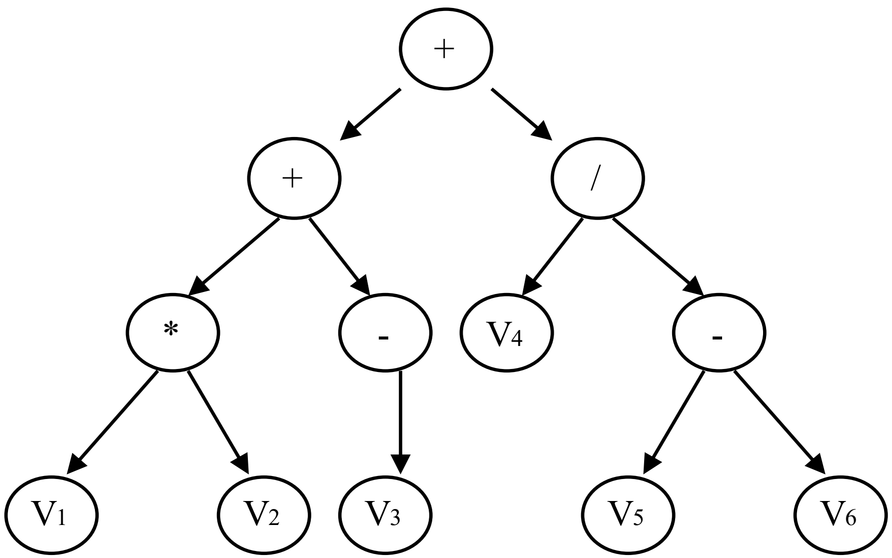
(ii) 如果有向森林F的每个弱分支都是有序树，并且也为F的每个弱分支规定了次序，则称F为有序森林。

- 在画有序树时，总是把根画在上部，并规定同一级上结点的次序是从左至右。
- 在画有序森林时，弱分支的次序也是从左至右。

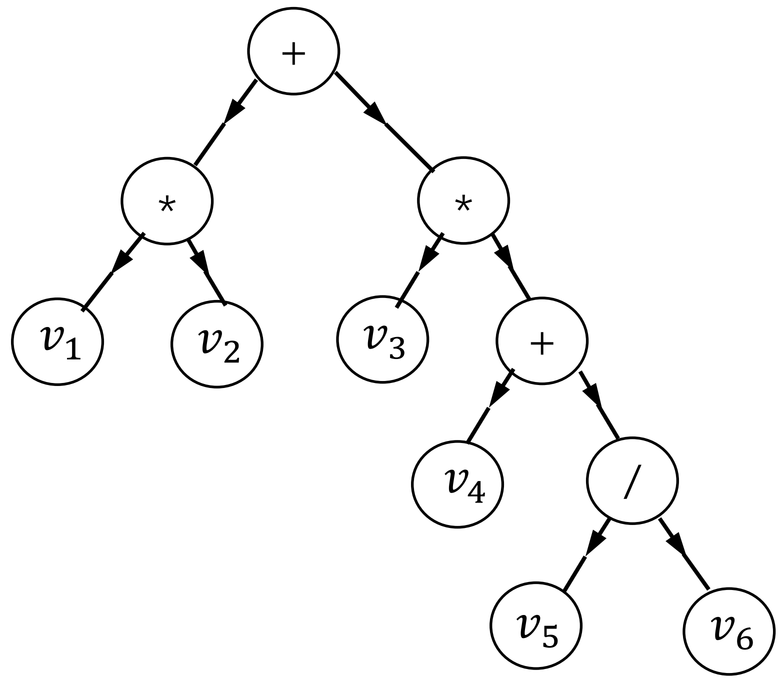
例：可以用有序树表示算术表达式，其中叶表示参加运算的数或变量，分支结点表示运算符。

如代数式 $((v_1*v_2)+(-v_3))+v_4/(v_5-v_6)$可表示为图G的有序数。

$v_1*v_2+v_3*(v_4+v_5/v_6)$ 可表示为图G’的有序树。

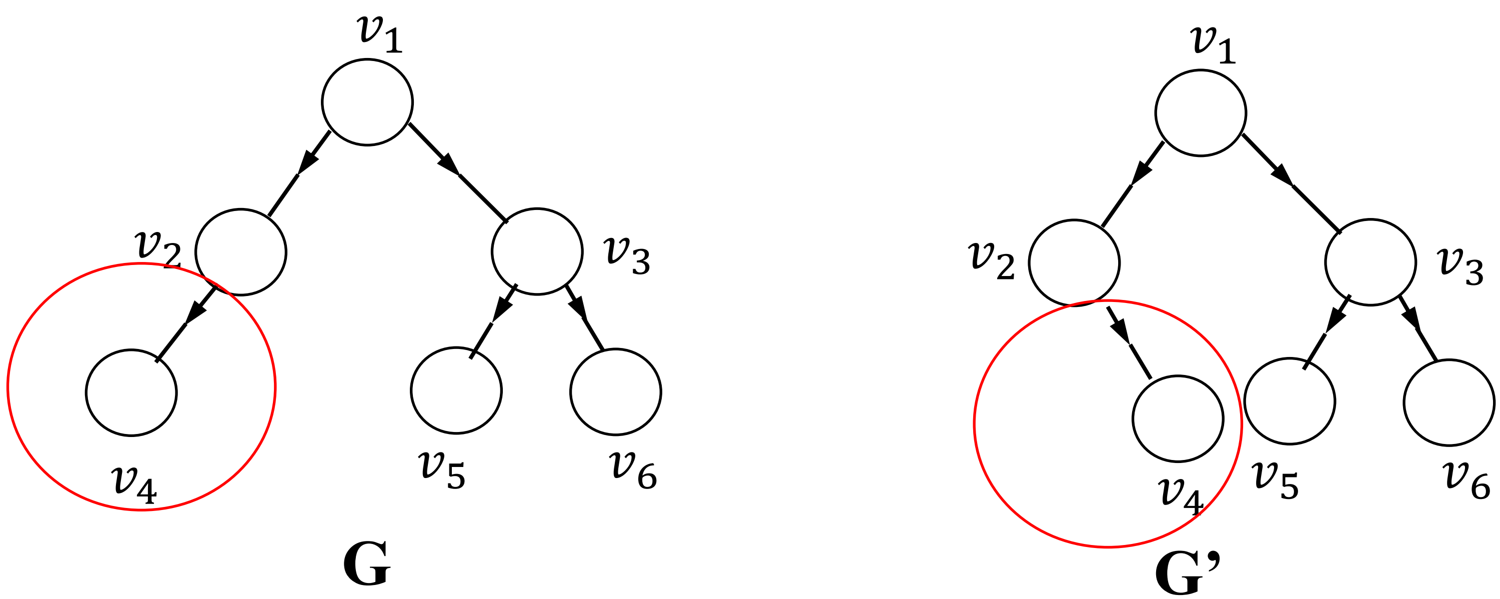


G



G’

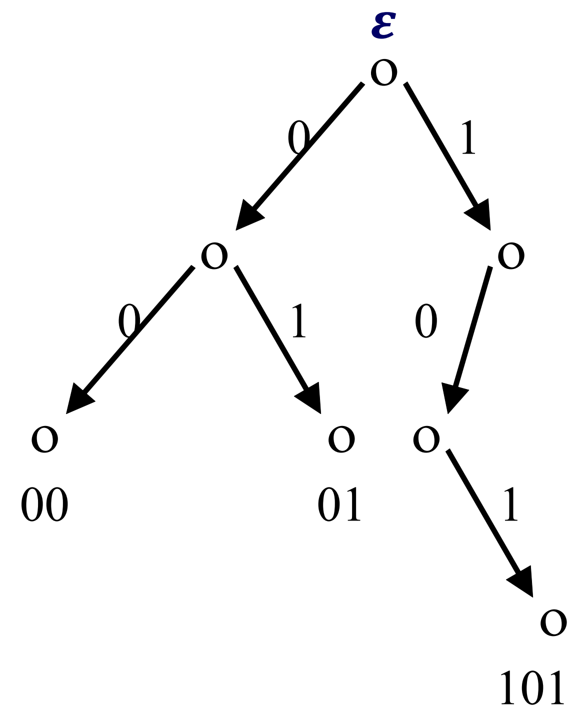
定位有序树



- G与G’是相同的有序树，因为同一级上结点的次序相同。
- 如果考虑结点之间的相对位置，G与G’不相同
- G与G’是不同的定位有序树

定位有序树

定义7.6.15 为每个分支结点的儿子规定了位置的有序树称为定位有序树。



例：在定位二元有序树中，可用字符表{0, 1}上的字符唯一地表示每个结点表示二进制编码情况。

1) 用空子府串ε表示根；
2) 设用 β表示某分支结点，则用β0表示它的左儿子，用β1表示它的右儿子。

这样，每个结点都有了唯一的编码表示，并且不同结点的编码表示不同。

定位二元有序树全体叶的编码表示的集合称为它的前缀编码

例：在计算机通信中要传输A, B, C, D, E, F, G, H八个字母，他们出现频率为A:30%, B:20%, C:15%, D:10%, E:10%, F:6%, G:5%, H:4%。给出一个最佳编码，使得通讯中出现的二进制数字尽可能少。

分析：

- 用较短（长）的序列去表示出现频率高（低）的字母，

问题转化为：

- 求出叶的权分别为0.04，0.05，0.06，0.1，0.1，0.15, 0.20, 0.3的最优二叉树，然后用这样的二叉树产生前缀编码传输上述给定的字母。
- **Huffman编码**
  1) 给定字母集C={c1, c2,…, cn}及频率f(c1),…, f(cn)；
  2) 设有n个叶结点，分别以f(c1),…, f(cn)为权；
  3) 在所有入度为0的结点，选出两个权最小的结点v, v’，添加一个新的分支节点u，使得u 以v和v’为儿子结点，且 f(u)=f(v)+f(v’)；
  4) 重复3)直至只有一个入度为0的结点。

Huffman 算法

- 建立Huffman树的主要运算是插入和删除最小频度字符，所以用最小堆。

算法HUFFMAN

输入：n个字符的集合$C=\{c_1,c_2,\ldots, c_n\}$及频率 $\{f(c_1), f(c_2),\ldots, f(c_n)\}$。

输出：C的Huffman树（V，T）。

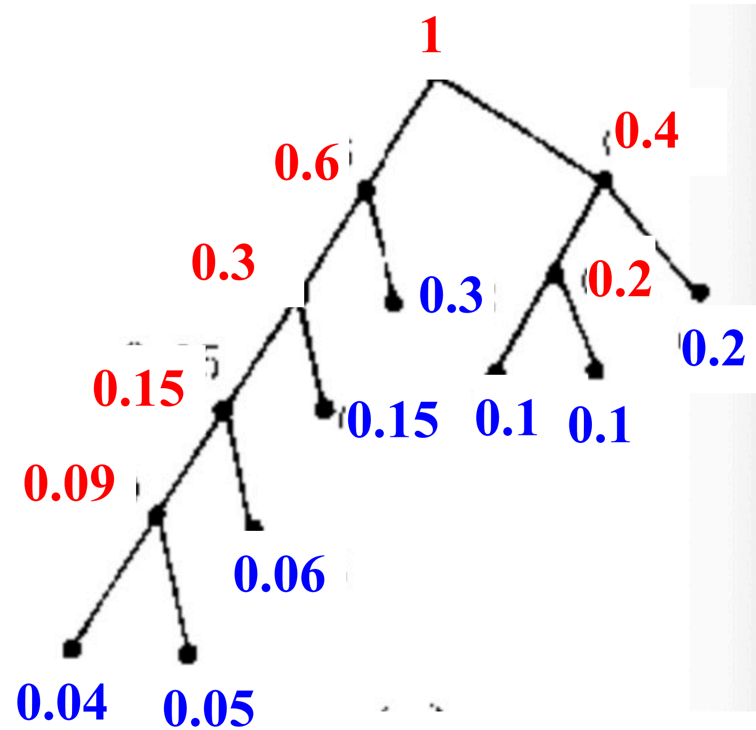
1.根据频度将所有字符插入最小堆H
2.$V \leftarrow C$； $T=\{\}$
3.for $j \leftarrow 1$ to $n-1$
4. $c \leftarrow$ DELETEMIN(H)
5. $c' \leftarrow$ DELETEMIN(H)
6. $f(v) \leftarrow f(c)+f(c')$ //新节点v
7. INSERT (H, v)
8. $V = V \cup \{v\}$
9. $T = T \cup \{(v, c), (v, c')\}$ // c, c'为T中v的孩子
10.end for

例：在计算机通信中要传输A, B, C, D, E, F, G, H八个字母，他们出现频率为A:30%, B:20%, C:15%, D:10%, E:10%, F:6%, G:5%, H:4%。给出一个最佳编码，使得通讯中出现的二进制数字尽可能少。

问题转化为：

- 求出叶的权分别为0.04，0.05，0.06，0.1，0.1，0.15, 0.20, 0.3 的最优二叉树，然后用这样的二叉树产生前缀编码传输上述给定的字母。

0.04	0.05	0.06	0.1	0.1	0.15	0.20	0.3
	0.09	0.06	0.1	0.1	0.15	0.20	0.3
		0.15	0.1	0.1	0.15	0.20	0.3
		0.15		0.2	0.15	0.20	0.3
				0.2	0.3	0.20	0.3
					0.3	0.40	0.3
						0.40	0.6
							1.0



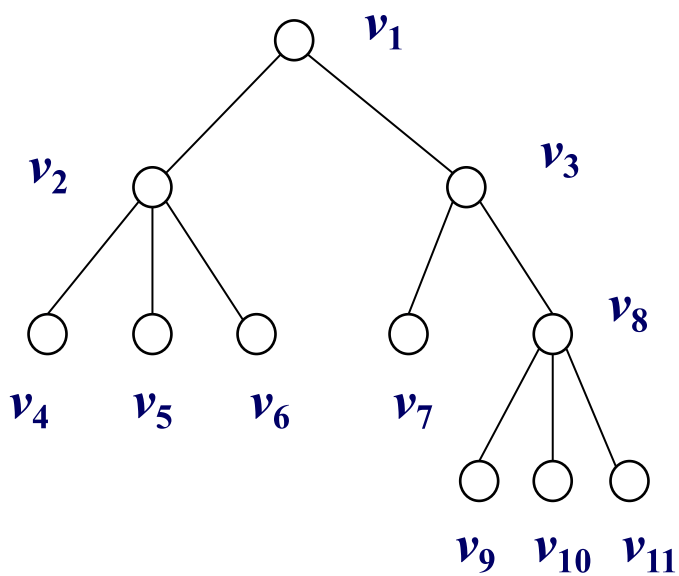
最佳编码

A: 01, B: 11, C: 001, D:100, E:101,
F: 0001, G: 00000, H:00001

有序树

- 借用家族树的名称来称呼有序树的结点。

称v_1是v_2和v_3的父亲，v_2是v_1的长子，v_2是v_3的哥哥，v_6是v_5的弟弟等等。



- 可以用定位二元有序树 表示有序森林。
- 有序森林和定位二元有序树之间建立一一对应关系。
  - 称位于左边的有序树之根为位于右边的有序树之根的哥哥

有序森林和定位有二元有序树之间的自然对应关系：

规定：**F与T有相同的结点。**

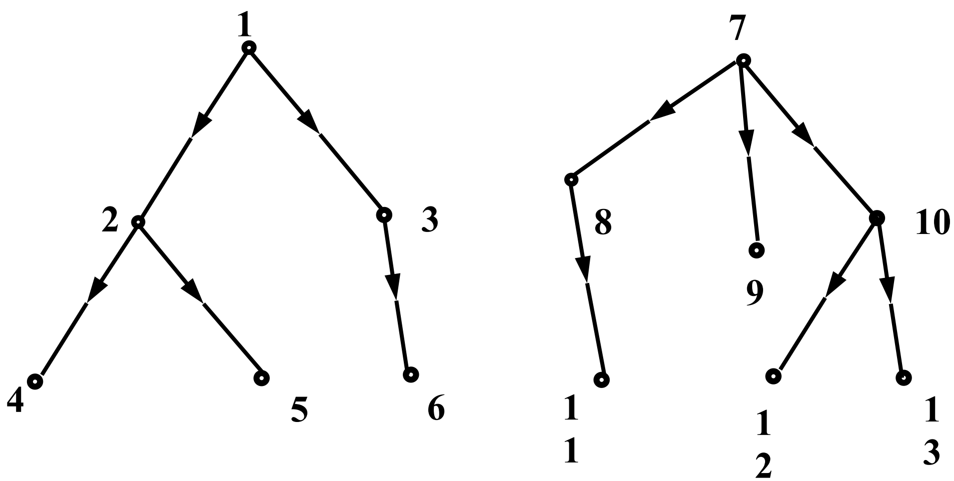
有序森林 F	定位二有序树 T
v_1是v_2的长子	v_1是v_2的左儿子
v_1是v_2的大弟	v_1是v_2的右儿子

亲弟弟，不包括堂兄弟

定位有序树

有序森林和定位二元有序树之间可以建立一一对应关系。

7是1的大弟



(a)

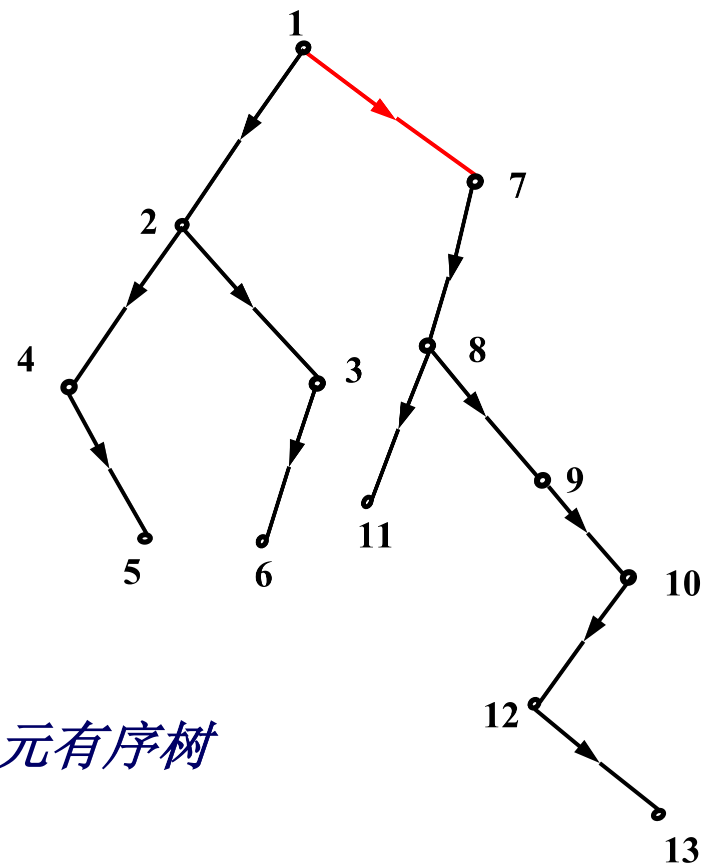


图7.6.11 有序森林和对应的定位二元有序树

一、判断题 （每小题 2 分，共 20 分）

（√） 1. 若$(A \oplus B) \cup C = \varnothing$，则 $A = B$。

（√） 2. 若 $A \subseteq C$ 且 $B \subseteq C$，则 $A \cup B \subseteq C$。

（×） 3. 若集合 A，B，C 和 D 满足 $A \times B \subseteq C \times D$，则 $A \subseteq C$ 且 $B \subseteq D$。

（×） 4. 设集合 A 上的二元关系 R_1 和 R_2 为传递的，则 $R_1 o R_2$ 也是传递的。

（×） 5. 若 R 是集合 A 上的二元关系，则 **rst** (R) 是 A 上的等价关系。

（×） 6. 设 $\le$ 为$\{a, b\}^*$中字符串的**字典序**，则$<\{a, b\}^*, \le>$ 是良序结构。

注：所谓**字典序**是指：$a \le b$，$a \le aa$，$abb \le baa$。

（√） 7. 设 $\boldsymbol{R}$ 为实数集，则 $\boldsymbol{R} \times \boldsymbol{R}$ 与 $\boldsymbol{R}$ 等势。

（√ ） 8. 设 f 是从 A 到 A 的满射且 $f \circ f = f$，则 $f = \mathbf{I}_A$。

（√ ） 9. 若无向图 G 中任意结点 v 的度数 $d_G(v) \ge 2$，则 G 中必存在回路。

（√ ） 10. n 阶二叉树有 $(n+1)/2$ 个叶结点。

二、设 A = {a，b，c，d} 上的二元关系 R_1 和 R_2 定义如下：（20 分）

$$R_1 = \{ \langle a, b\rangle, \langle b, a\rangle, \langle a, c\rangle, \langle c, a\rangle \}$$

$$R_2 = \{ \langle a, b\rangle, \langle b, c\rangle, \langle c, a\rangle \}$$

1） 试分别指出 R_1 和 R_2 所具有的性质（即是否具有自反性、反自反性、对称性、反对称性和传递性这五种性质）。

2） 试求出 R_1^2， $R_1 \circ R_2$ 和 R_2^+。

三、设$\#(A) \geq 2$，$\boldsymbol{\rho}(A)$ 为 A 的幂集，$\subseteq$ 为子集关系。（15 分）

1）**证明**： $<\boldsymbol{\rho}(A), \subseteq>$ 是偏序结构；

证明：分别证明 $\subseteq$ 是 $\boldsymbol{\rho}(A)$上的自反、反对称、传递的二元关系。（5 分）

a) 对任意 $B \in \boldsymbol{\rho}(A)$：$B \subseteq B$，即 $\subseteq$ 是自反的；

b) 对任意 $B, C \in \boldsymbol{\rho}(A)$，若 $B \subseteq C$ 且 $C \subseteq B$，则 $B = C$，即 $\subseteq$ 是反对称的；

c) 对任意 B, C，$D \in \boldsymbol{\rho}(A)$，若 $B \subseteq C$ 且 $C \subseteq D$，则 $B \subseteq D$，即 $\subseteq$ 是传递的。

因此，$<\boldsymbol{\rho}(A), \subseteq>$ 是偏序结构。

2）**说明**：$<\boldsymbol{\rho}(A), \subseteq>$ 不是全序结构；（5 分）

因为$\#(A) \geq 2$，设 b, c 为 A 中的不同元素，则$\{b\}$，$\{c\} \in \boldsymbol{\rho}(A)$，且$\mathbf{\{b\}} \subseteq \mathbf{\{c\}}, \mathbf{\{c\}} \subseteq \mathbf{\{b\}}$ 均不成立，故$<\boldsymbol{\rho}(A), \subseteq>$ 不是全序结构。

3）任给 B, C∈**ρ** (A)，求出集合{B, C}的最小上界与最大下界，并**加以证明**。（5 分）

证明：任给 B, C ∈ **ρ** (A)：$\boldsymbol{B\cup C}$、$\boldsymbol{B\cap C}$ 分别为{B, C}的**上确界**和**下确界**。

a) 若 D 为{B, C} 的**上界**，即 $B\subseteq D$，$C\subseteq D$，则 $\boldsymbol{B\cup C\subseteq D}$；

且 $\boldsymbol{B\cup C}$ **本身为 {*B*，*C*} 的上界**（因 $B\subseteq B\cup C$，$C\subseteq B\cup C$），

故 $B\cup C$ 为{B, C}的最小上界，即：$B\cup C$ 为{B, C}的上确界。

b) 若 E 为{B, C} 的**下界**，即 $E\subseteq B$，$E\subseteq C$，则 $E\subseteq B\cap C$；

且 $B\cap C$ 本身为{B, C}的下界（因 $B\cap C\subseteq B$，$B\cap C\subseteq C$），

故 $B\cap C$ 为{B, C}的最大下界，即：$B\cap C$ 为{B, C}的下确界。

四、试求叶的权分别为 2，5，10，17，26，37，50，65 的**最优叶加权二叉树**及其**叶加权路径长度**。（12 分）

五、设 f：X →Y 为全函数。**证明**；f 为**右可逆的**，当且仅当 f 为**满射**。（12 分）

证明：

⇒）若 f 为**右可逆的**，则有 g：$Y \to X$ 使 $f \circ g = \mathbf{I}_Y$，

从而由书上定理（**左满右单**）可知：f 为 满射。

⇐）另一方面，设 f 为 **满**射。

i) 当 $X = \varnothing$ 时，因 f 为满射，则 $Y = \mathrm{ran}\, f = \varnothing$，定理显然成立。

ii) 当 $X \neq \varnothing$ 时，因 f 为全函数，则 $Y \neq \varnothing$。对每个 $y \in Y$，令

$$\mathbf{S}_y = \{ x \mid x \in X \text{ 且 } f(x) = y \}$$

则 $\{ \mathbf{S}_y \mid y \in Y \}$ 就是 X 的一个划分。对每个 $y \in Y$，都任意取定 $\mathbf{S}_y$ 中唯一的一个元素 $\mathbf{x}_y$，显然 $f(x_y) = y$。并令：

$$g = \{ < y, \mathbf{x}_y > \mid y \in Y \}$$

则 g 显然是一个从 Y 到 X 的全函数，且 $f \circ g = \mathbf{I}_Y$ 。

这表明 g 是 f 的一个右逆，即 f 为**右可逆的**。

六、设 n 阶**连通无向图** G 为**非循环的**，直接用**归纳法**证明：G 有 n – 1 条边。（9 分）

证明：**施归纳于 n：**

当 n =1 时，由 G **非循环**可知：G 没有自圈，即 G 有 0 条边，故命题为真。

假设对任意 $k\geq1$，当 n =k 时命题为真。

当 n = k+1 时：因 G 为**连通的**，故任意结点 v 的度数 $d_G(v) \geq 1$。

若 G 中任意结点 v 的度数 $d_G(v) \geq 2$，则 G 中必存在回路，这与 G 为**非循环**的条件**矛盾！**因此，G 中必有**结点** v_1 的度数 $d_G(v_1)=1$。

显然，k 阶无向图 $G - v_1$ 连通且非循环，由**归纳假设** $G - v_1$ 有 k – 1 条边。设与 v_1 相邻的结点为 v_2，v_1 与 v_2 的连接边为 e，G 可由 $G - v_1$ 添加结点 v_1 与连接边 e 得到，所以 G 有 k 条边，即 n =k+1 时命题亦为真。

综上所述，命题为真。

七、设 R 为集合 A 上的二元关系，证明 $\mathbf{t}(R)=R^+$ 。（12 分）

其中，$\mathbf{t}(R)$ 为 R 的**传递闭包**，$R^+ = \bigcup_{i=1}^{\infty} R^i$。

证明：

（一）**先证** $t(R)\subseteq R^+$：

若<x，y>，<y，z>$\in R^+$，则必有 n，m$\in I_+$ 使 <x，y>$\in R^n$ 且<y，z>$\in R^m$，因此<x，z>$\in R^{n+m}$，所以<x，z>$\in R^+$。 即 R^+ 是**传递的**。

由 $R\subseteq R^+$ 以及**传递闭包**的定义可知： $t(R)\subseteq R^+$ 。

七、设 R 为集合 A 上的二元关系，证明 $\mathbf{t}(R)=R^{+}$ 。（12 分）

其中，$\mathbf{t}(R)$ 为 R 的**传递闭包**，$R^{+}=\bigcup_{i=1}^{\infty}R^{i}$。

（二）**再证** $R^{+}\subseteq t(R)$，为此只需证明：

对每个 $\mathbf{n}\in\mathbf{I}_{+}$，皆有 $R^{n}\subseteq t(R)$。

这可用**归纳法证明**如下：

1) 根据闭包的定义可知，$R\subseteq t(R)$，即当 n=1 时命题为真；

2) 对任意的 $k\in I_{+}$，假定当 $n=k$ 时命题为真，即 $R^{k}\subseteq t(R)$。

任取$<x, z>\in R^{k+1}$，因为 $R^{k+1}=R\circ R^{k}$，所以有 $y\in A$ 使得 $<x, y>\in R$ 且$<y, z>\in R^{k}$。

由 $R\subseteq t(R)$ 且 $R^{k}\subseteq t(R)$ 可知：$<x, y>\in t(R)$ 且$<y, z>\in t(R)$。

而 $t(R)$ 又是传递的，故 $<x, z>\in t(R)$。

这表明 $R^{k+1}\subseteq t(R)$，即当 n=k+1 时命题也真。

因此，**对每个 $\mathbf{n}\in\mathbf{I}_{+}$，皆有** $R^{n}\subseteq t(R)$。故有 $R^{+}\subseteq t(R)$。

因此，必有 $t(R)=R^{+}$。