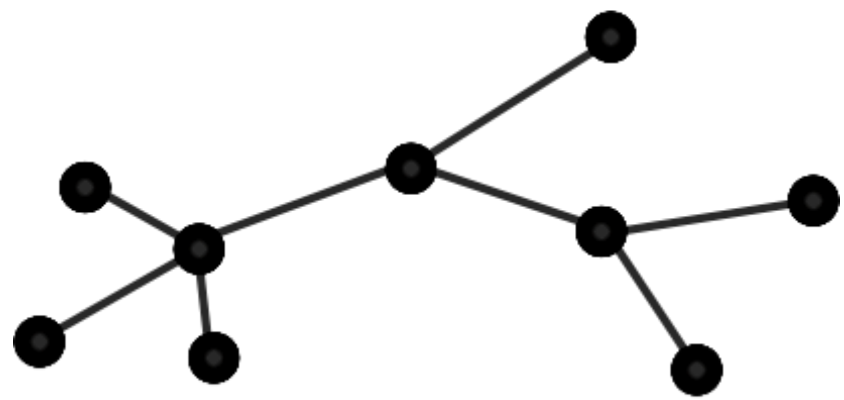
无向树的定义

定义　一个连通且无回路的无向图称为树。

树叶：树中度数为1的顶点；

分枝点：度数大于1的顶点

森林：每个连通分支都是树的无向图。



树的6个等价定义

定理　给定图T，以下关于树的定义是等价的。

（1）无回路的连通图;

（2）无回路且$e=v-1$，其中e是边数，v是顶点数;

（3）连通且$e=v-1$;

（4）无回路，但增加一条新边，得到一个且仅有一个包含新边的回路。

（5）连通且每条边均为桥;

（6）G中任意两个顶点之间存在惟一的路径;

(1)⇒(2)的证明

如果T是无回路的连通图，则G中无回路且$e=v-1$，其中e是边数，v是顶点数

证明 归纳法。

当$v=2$时，因为T连通无回路，

所以只有$e=1$，故$e=v-1$成立。

假设$v=k-1$时命题成立，当$v=k$时，

因T是无回路且连通，则至少有一个度为1的顶点u，

设与其关联的边为(u,w)，删去u，得到一个$k-1$个顶点

的连通无向图T'，

(1)⇒(2)的证明（续）

由归纳假设可知，　T'的边数$e'=v'-1=(k-1)-1=k-2$。

再将顶点u及(u,w)放入原位，恢复到图T，

那么T的边数

$e=e'+1=(k-2)+1=k-1$，

顶点数$v=v'+1=k$，

故$e=v-1$成立。

(2)⇒(3)的证明

如果T中无回路且$e=v-1$，其中e是边数，v是顶点数，则连通且$e=v-1$;

只须证明T是连通的。

证明 设T有k个连通分枝$T_1,\ldots,T_k(k\geq 2)$，T_i有v_i个顶点,e_i条边，因为T_i连通无回路，所以有

$$e_i=v_i-1,\quad v=v_1+v_2+\ldots+v_k$$

$$e=e_1+e_2+\ldots+e_k=(v_1-1)+(v_2-1)+\ldots+(v_k-1)=v-k$$

因为$e=v-1$，所以$k=1$，故T是连通的。

(3)⇒(4)的证明

如果T连通且$e=v-1$，则T中无回路，但增加一条新边，得到一个且仅有一个包含新边的回路。

证明 归纳法。

当$v=2$时，$e=v-1=1$，必无回路，如果增加一边得到且仅得到一个回路。

设$v=k-1$时命题成立。考察$v=k$时的情况。

因为T是连通的，所以每个顶点u有$\deg(u)\geq 1$，

下面证明至少有一个顶点u_0使$\deg(u_0)=1$。

若不存在，则每个顶点的度至少为2，所以$2v\geq 2e$，即$v\geq e$，这与$e=v-1$矛盾。

(3)⇒(4)的证明

删去u_0及其关联的边，得到含有k-1个顶点的图T'，
T'连通且$e'=v'-1$。由归纳假设知T'无回路。
在T'中加入u_0及其关联的边恢复到T，则T无回路。
若在T中增加一条边(u_i,u_j)，
因为T连通，则在T中存在一条从u_i到u_j的路，
那么这条路与新加入的边(u_i,u_j)构成回路，
而且这个回路是唯一的。
若不唯一，删掉边(u_i,u_j)边,T中必有回路，矛盾。

(4) ⇒(5)的证明

如果T中无回路, 但增加一条新边，得到一个且仅有一个包含新边的回路，则T连通且每条边均为桥。

证明 反证法。

假设T不连通,

则存在顶点u_i与u_j，在u_i和u_j之间没有路,

所以增加边(u_i,u_j)不会产生回路，与已知矛盾。

由于T无回路，故删掉任意条边e都使T-e为非连通,

所以T中每条边都是桥。

(5) ⇒(6)的证明

如果*T*连通且每条边均为桥，则*T*中任意两个顶点之间存在惟一的路径。

证明　由T是连通的可知，任意两个顶点间有一条路，

若存在两点它们之间有多于一条的路，

则T中必有回路，

删去该回路上任一边，

图仍是连通的，

与T中每条边都是桥矛盾。

(6) ⇒(1)的证明

如果*T*中任意两个顶点之间存在惟一的路径，则*T*是无回路的连通图。

证明　因为任意两顶点间有唯一条路，则图*T*必连通。

若*T*有回路，

则在回路上任意两顶点间有两条路，

与已知矛盾。

无向树的性质

定理 任一棵树中至少有两片树叶。

证 设$T=<V,E>$是无向树，其中$|V|=v$，$|E|=e$

设T有x片树叶，则剩余的$v-x$各顶点的度均大于等于2，

由握手定理及定理7-7.1可知，

$$2(v-1)=\sum d(v_i)\geq x+2(v-x)$$

解上式可得 $x\geq 2$.

定理得证。

关于无向树计算的实例

例　一棵无向树T有5片树叶，3个2度分支点，其余的分支点都是3度顶点，问T有几个顶点。

解　设有n个顶点，则

$$5+3\times2+(n-8)\times3=2(n-1)$$

得$n=11$

实例

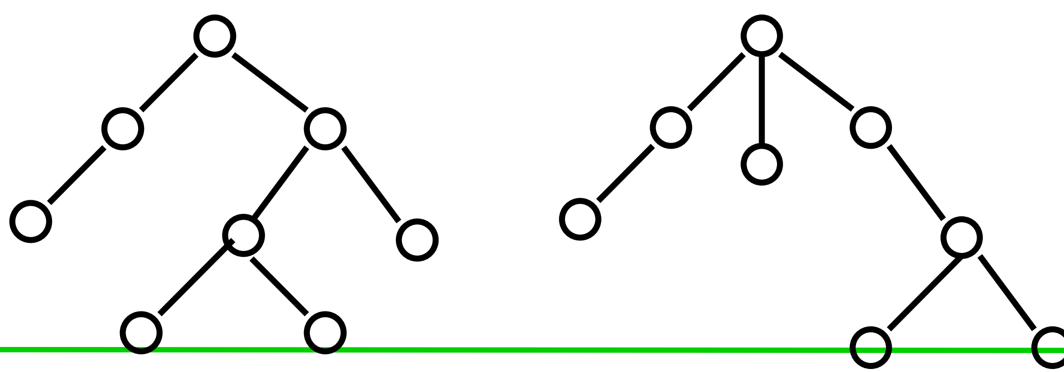
例　下面两个正整数序列中，哪个能充当无向树的度数序列？若能,画出2棵非同构的无向树。

（1）4,3,3,2,1,1,1,1

（2）3,3,2,2,1,1,1,1

解　（1）不可以，因为所有度数之和等于16，而顶点数为8，假设可以构成树，则度数之和应为14，所以不可以。

（2）可以。



生成树

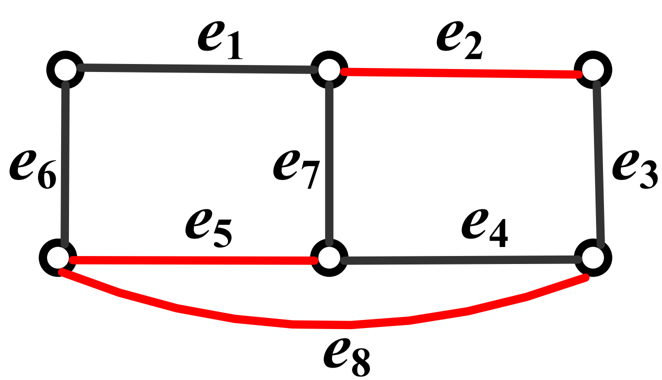
定义　若图G的生成子图是一棵树，则该树称为G的生成树。

生成树T的树枝: G在T中的边

生成树T的弦: G不在T中的边

举例

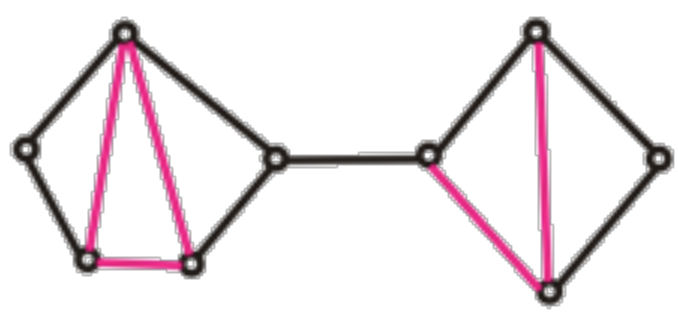
例　如下图，$T=\{e_1,e_6,e_7,e_4,e_3\}$，



$\overline{T}=\{e_2,e_5,e_8\}$

余树是非连通的，无回路

黑线表示生成树，红线构成树的补。



余树是非连通的，有回路

生成树的存在性定理

定理 连通图至少有一棵生成树。

证 用破圈法.

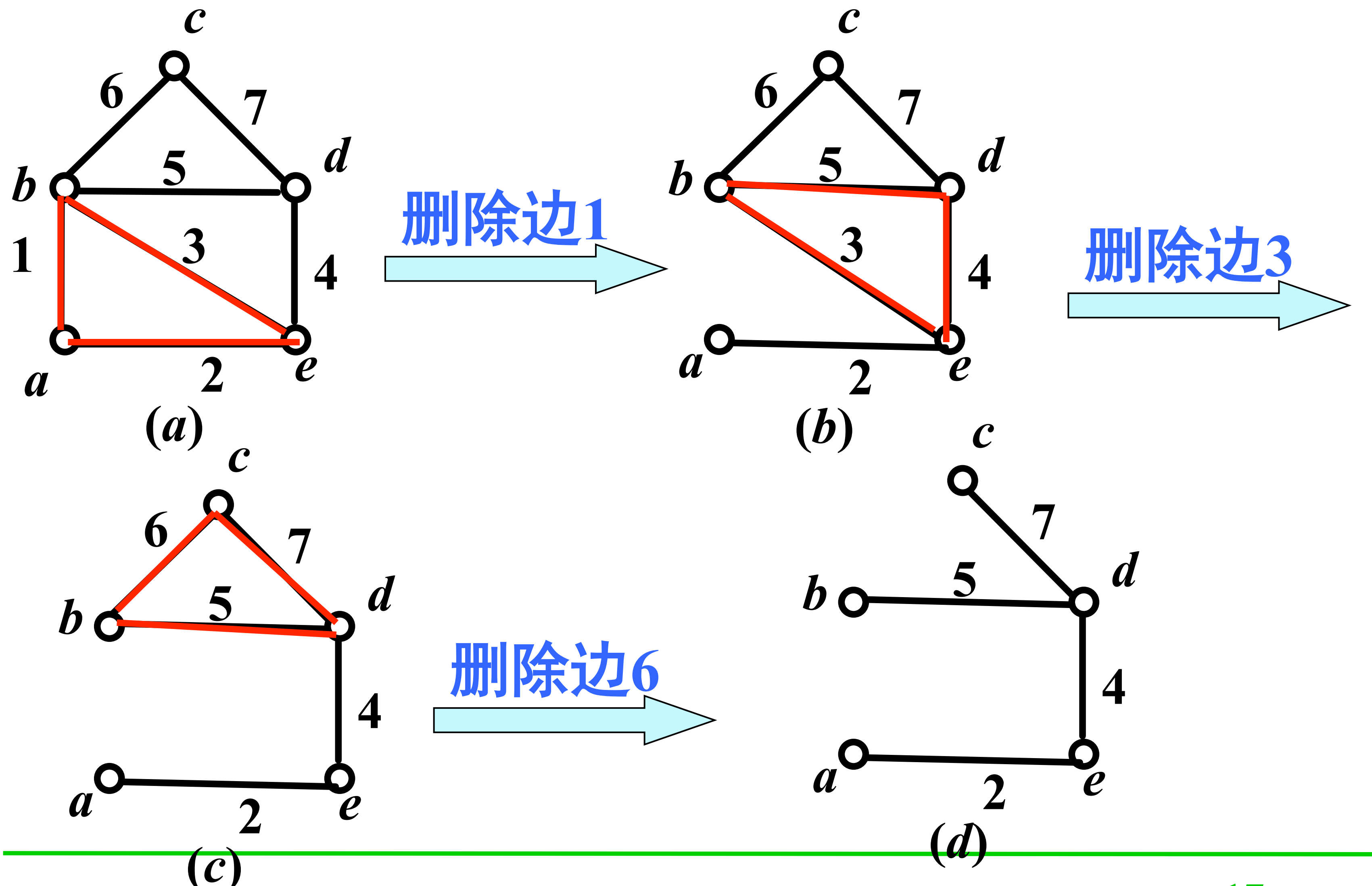
若图中无圈, 则图本身就是生成树.

否则删去圈上的任一条边, 这不破坏连通性,

重复进行直到无圈为止,剩下的图是一棵生成树.

注意: 连通图的生成树不唯一。

举例



几个结论

- 设n阶无向连通图有m条边, 则$m \geq n-1$.
- 设n阶无向连通图有m条边, 则它的生成树的补有$m-n+1$条边. $m-n+1$称为连通图G的秩。

小练习：

含有n个顶点且至少有n条边的无向简单图，必有回路。

最小生成树

设G是具有n个顶点的连通图，对于G的每一条边e指定一个正实数$C(e)$，称作边e的权．图连同附加在边上的权称作带权图，记作$G=\langle V,E,C\rangle$.

设G'是G的子图,

G的权$C(G')$：G'所有边的权之和称作。

设T是G的生成树,

树权$C(T)$：T的所有边权之和。

定义　在图G的所有生成树中，树权最小的生成树，称作最小生成树。

最小生成树的算法

求最小生成树的算法——避圈法 (Kruskal)

设$G=<V,E,C>$, 将边按权从小到大排序：$e_1, e_2, \ldots, e_m$.

(1) 取最小权边e_1，置边数$i \leftarrow 1$；

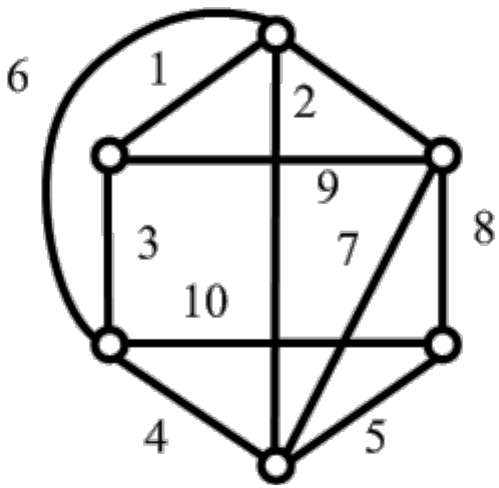
(2) $i=n-1$结束，否则转(3)；

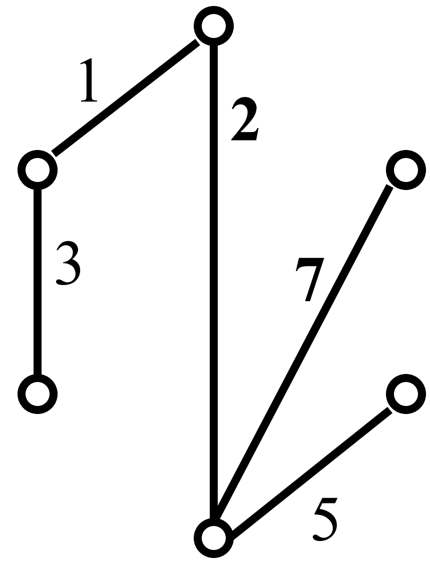
(3)设已选择边 $e_1, e_2, \ldots, e_i$，在G中选取不同于$e_1, e_2, \ldots, e_i$的边e_{i+1}，使$\{e_1, e_2, \ldots, e_i, e_{i+1}\}$中无回路且$e_{i+1}$是满足此条件的最小边。

(4) $i \leftarrow i+1$，转(2).

举例

例　求下面赋权图的最小生成树。





$C(T)=18$

根树的定义

定义　一棵有向树，如果恰有一个顶点的入度为0，其余所有顶点的入度都为1，则称为根树。

根：入度为零的顶点；

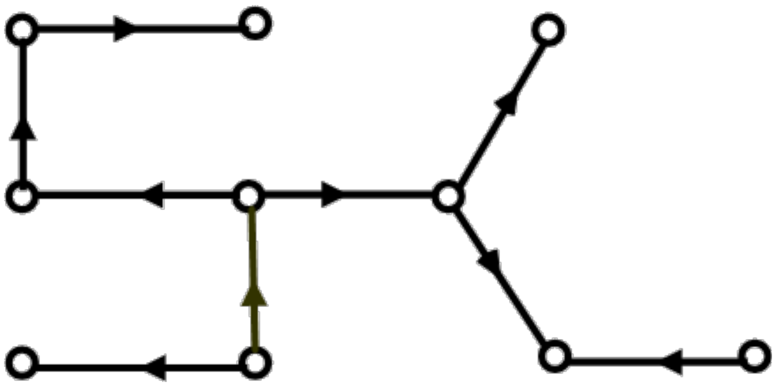
叶：入度为1, 出度为0的顶点

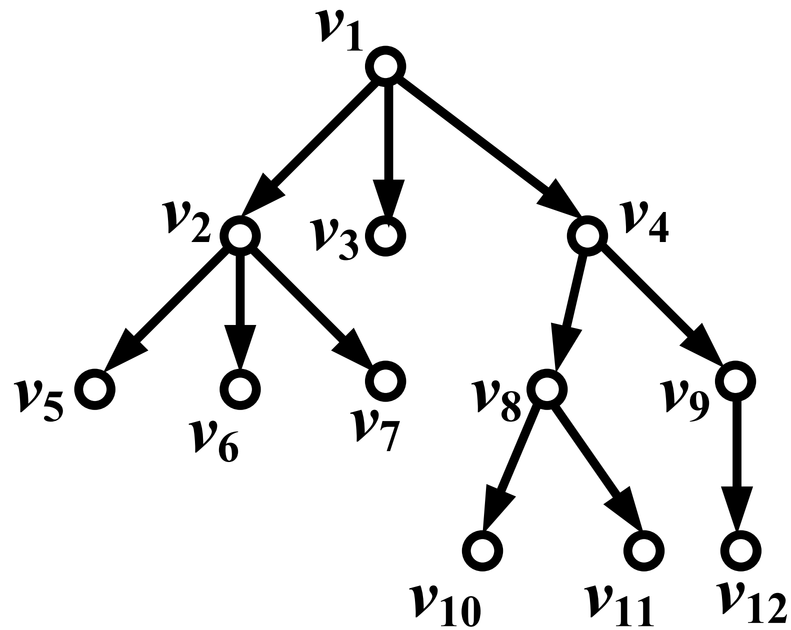
分枝点: 出度大于0的顶点

顶点v的层数: 从树根到v的通路长度

树高: 有向树中顶点的最大层数

举例





有向树
不是根树

根树
根： v_1
分支顶点： v_1, v_2, v_4, v_8, v_9
叶： $v_5, v_6, v_7, v_{10}, v_{11}, v_{12}$
树高： 3

家族树

定义　把根树看作一棵家族树:

(1) 若顶点 a到顶点 b有一条边, 则称 b 是 a 的儿子, a 是 b的父亲;

(2) 若b和c为同一个顶点的儿子, 则称b和c是兄弟;

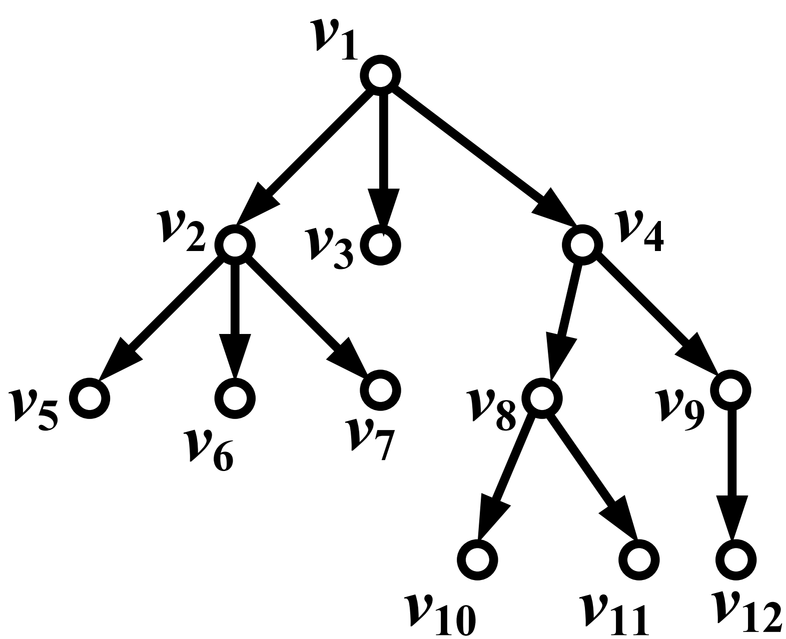
(3) 若$a \neq b$且a到b有一条单向通路, 则称a是b的祖先, b是a的后裔.

根树的递归定义

根子树：设v是根树的一个顶点且不是树根, 称v及其所有后代的导出子图为以v为根的根子树。

定义　根树中除树根v外，其它所有顶点被分成有限个子根树。

如右图所示



根树由分别以v_2,v_3,v_4为根的3个子根树组成。

以v_4为根的子树由分别以v_8,v_9为根的两个根子树构成。

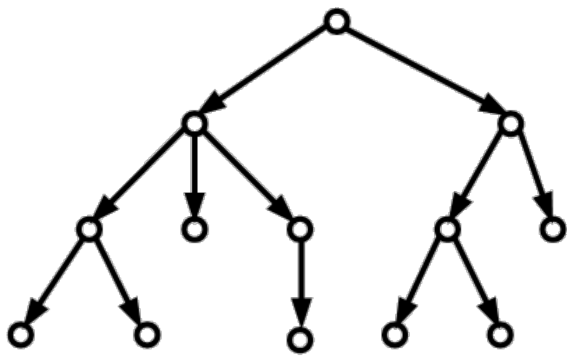
有序树

有序树：指明根树每一层次上顶点的次序。

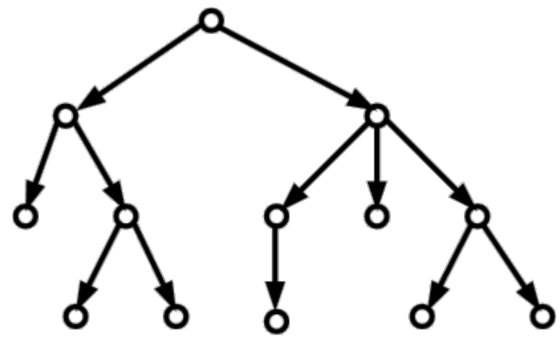
根树的画法：从根向上生长和由树根向下生长。

这时可省略边的

方向。



(b)



(c)

对于无序树，两图同构，
但对于有序树，则不然。

m叉树与完全树

定义　在根树中，若每一个顶点的出度小于等于m，则称这棵树为m叉树。

完全m叉树：每一个顶点的出度恰好等于m或零

正则m叉树：所有树叶层次相同的m叉树

二叉树：$m=2$

有序树可以改为二叉树

定理　任意一棵有序树可以改写为一棵对应的二叉树。

方法：

对于有序树中的每一个子树作如下处理：

设子树的根为u，

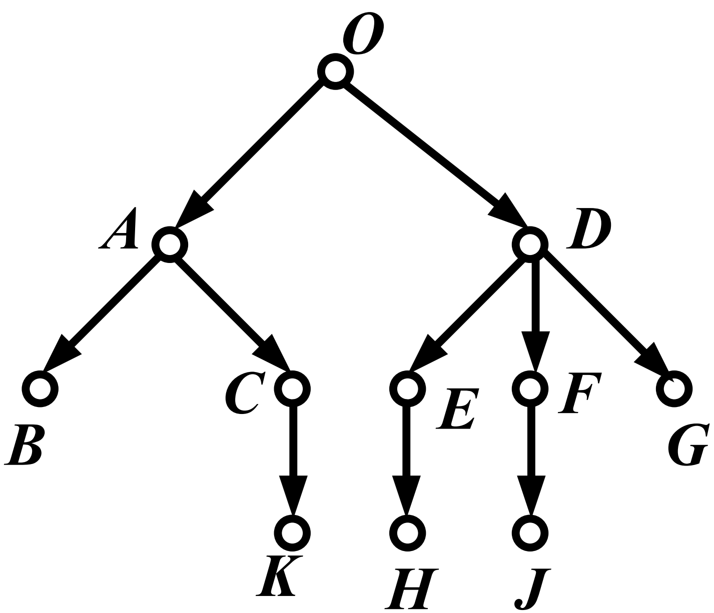
保留u的最左边儿子顶点作为二叉树中顶点u的左儿子，

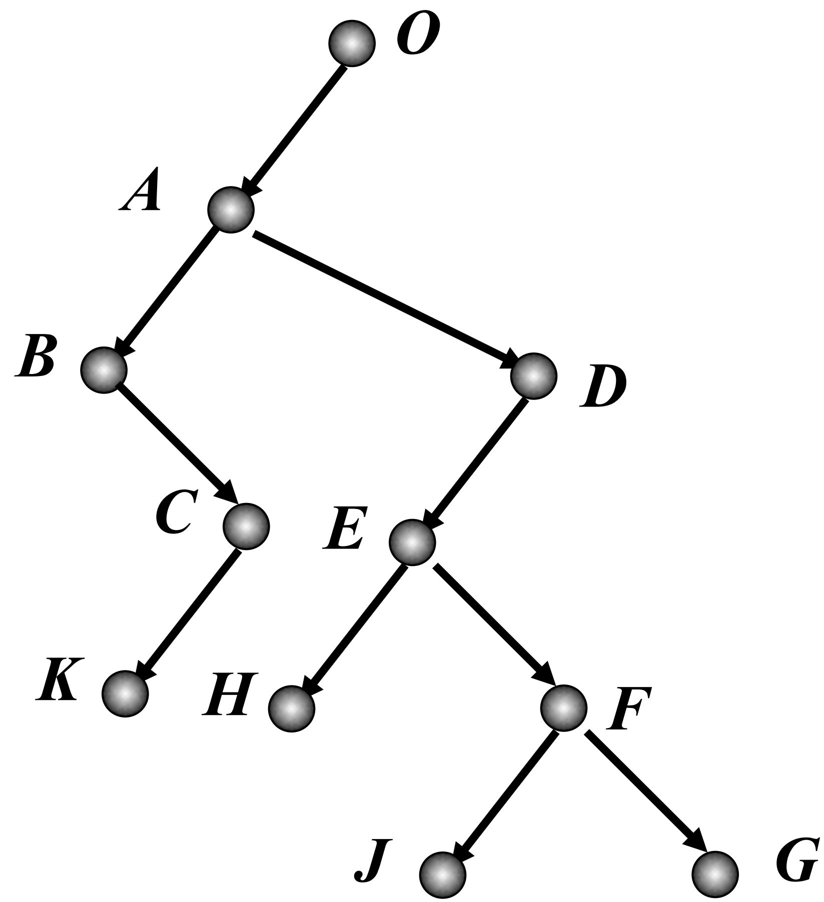
u的兄弟顶点作为u的右儿子，

若u无兄弟顶点，则u的右儿子为空。

举例

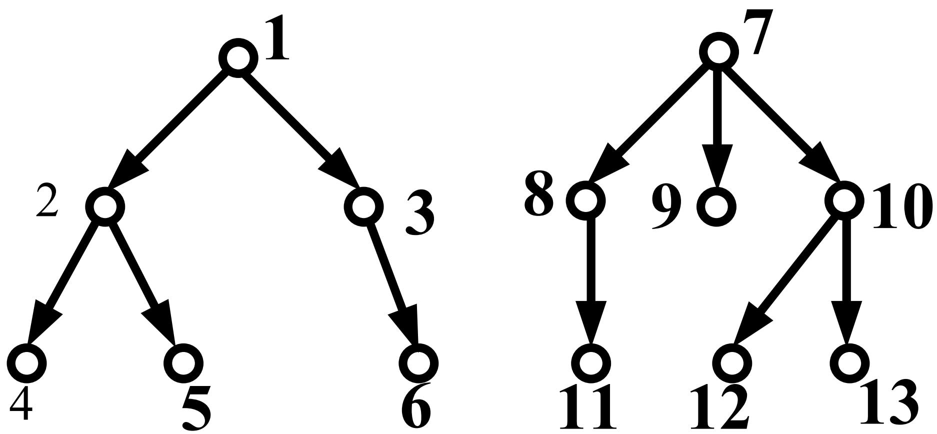
把下面的有序树改写为二叉树。

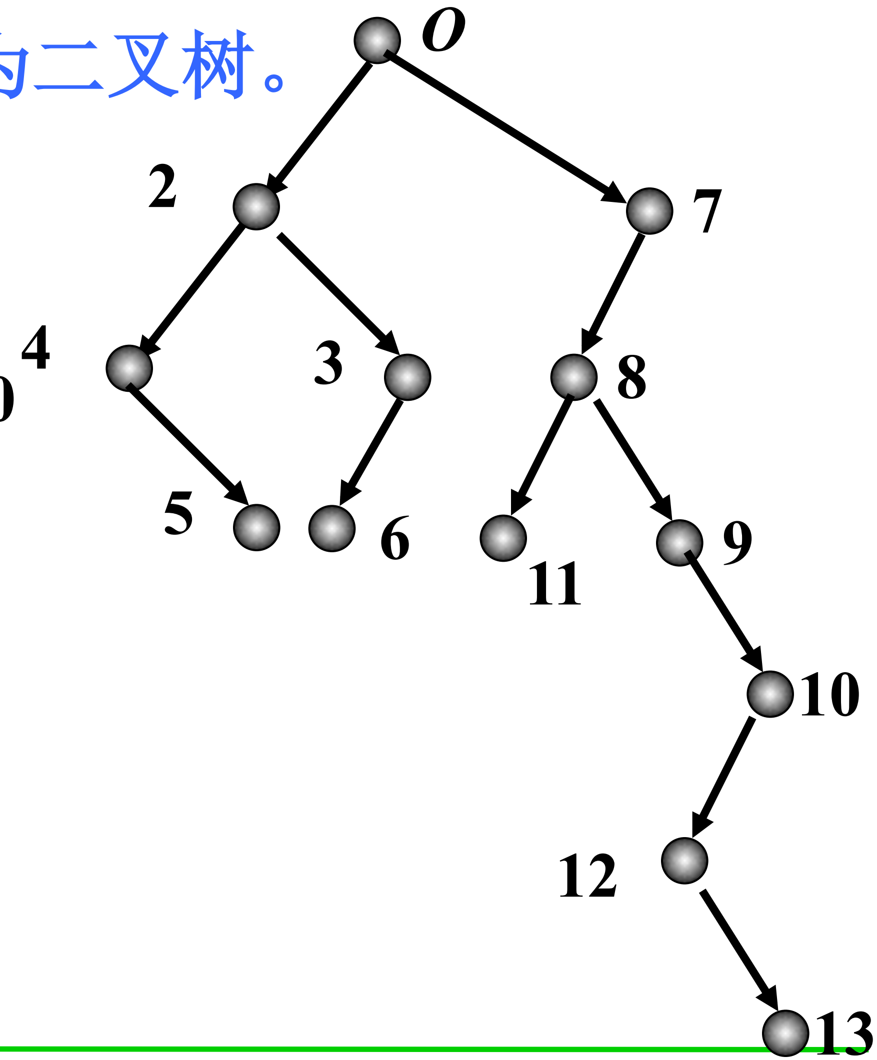




举例

把下面的有序森林改写为二叉树。

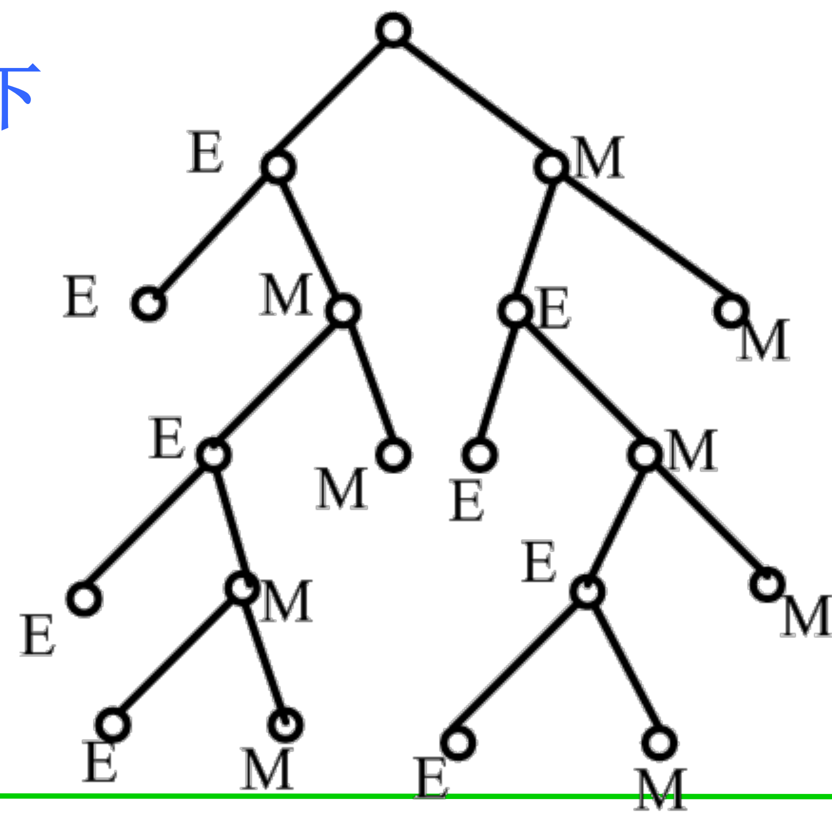




2叉树的应用实例

例　*M*和*E*两人进行网球比赛，如果一人连胜两盘或共胜三盘就获胜，比赛结束。表示出比赛的各种可能情况。

解：可以用根树表示如下



从根到树叶的每条路表示一种情况。
共有10片树叶，所以共有10种比赛情况。如
EE,EMM,EMEMM
EMEME

m叉树的性质

定理　设有完全m叉树，其树叶数为t，分枝点数为i，则$(m-1)i=t-1$。

证明　完全m叉树的每个分枝顶点的出度均是m，树叶的出度为零.

除了根顶点，其余每个顶点的入度都为1，

根据有向图的握手原理得

$$e=t+i-1=mi$$

即有 $(m-1)i=t-1$。

应用实例

例题 设有28盏灯，拟公用一个电源插座，问需用多少块具有四插座的接线板。

解 将每个四插座的接线板看作分枝顶点，每盏灯看作是树叶，可以构成完全4叉树，那么t=28，m=4

由$(m-1)i=t-1$，可得i=9.

应用实例

例题　假设一台计算机有一条加法指令，可计算3个数的和，如果要计算9个数的和，至少要执行几次加法指令。

解：把9个数看成完全3叉树的树叶，加法指令则是分枝点，所以

$$(3-1)i=9-1$$

$$i=4$$

故需要执行4次加法指令。

Huffman编码

通讯编码

- Shannon, Hamming
- 比特(bit): binary information unit
- 例: {0,1,2,…,7}, $\log_2 8=3$, 编码为

  000,001,010,…,111

例：000111010101译为0725

Huffman编码

不等长编码

- 若{0,1,2,…,7}出现频率不一样,则出现频率高的用短码字

例: 频率递减: 0,1,2,3,4,5,6,7, 编码为

0,1,00,01,10,11,000,001.

收到000111

不能唯一解码：651, 235, 075,…等.

- 原因: 码字互为前缀,如00是001的前缀

Huffman编码

前缀码(Prefix codes)

- 前缀码: 码字互相不为前缀的不等长编码

- 例:{0,1,2,3}编码为{00,010,011,1}
  收到000111,译为023

Huffman编码

最佳前缀码

- **最佳前缀码**: 给定信号出现频率, 平均码字长度最短的前缀码
- **平均码字长度**: 码字长度乘以频率,求和
- 例: {0,1,2,3}, 40%, 30%, 20%,10%

编码1: {00,010,011,1},

- $2\times40\%+3\times30\%+3\times20\%+1\times10\%=2.4$

编码2: {1,00,010,011},

- $1\times40\%+2\times30\%+3\times20\%+3\times10\%=1.9$

二叉树的应用——最优树

定义 给定一组权$w_1,w_2,\ldots,w_t$，不妨设$w_1 \le w_2 \le \ldots \le w_t$。设有一棵二叉树，共有$t$片树叶，分别带权$w_1,w_2,\ldots,w_t$，该二叉树称为带权二叉树。

带权二叉树的权：
$$w(T)=\sum_{i=1}^{t} w_i L(w_i)$$

其中$L(w_i)$是带权为w_i的树叶的通路长度。

最优树：在所有带权$w_1,w_2,\ldots,w_t$的二叉树中，$w(T)$最小的那棵树。

最优树的构造方法

Huffman算法:

给定实数$w_1, w_2, \ldots, w_t$,

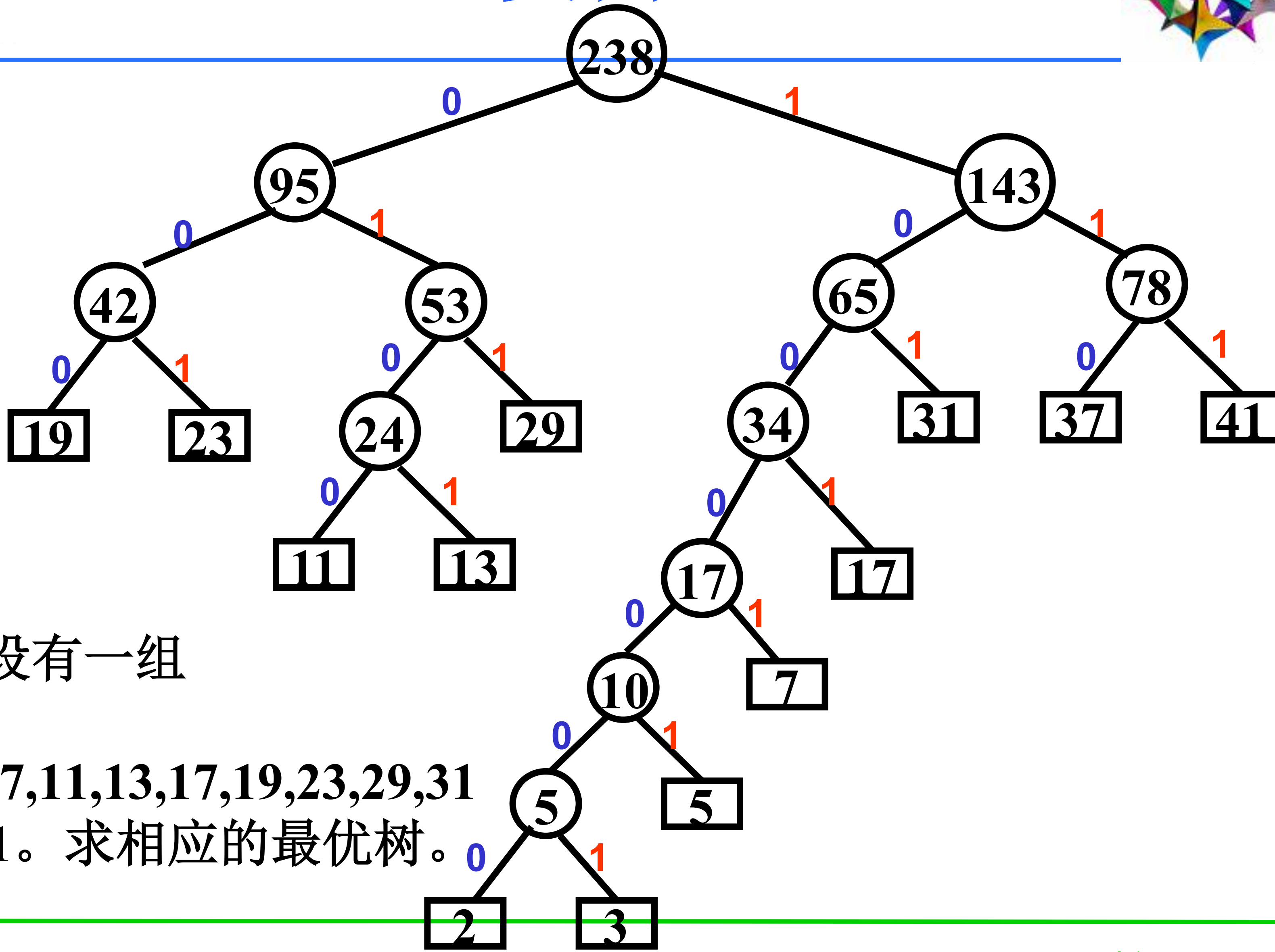
① 作t片树叶, 分别以$w_1, w_2, \ldots, w_t$为权.

② 在所有入度为0的顶点(不一定是树叶)中选出两个权最小的顶点，添加一个新分支点，以这2个顶点为儿子，其权等于这2个儿子的权之和.

③ 重复②, 直到只有1个入度为0 的顶点为止.

$W(T)$等于所有分支点的权之和

实例



例题3 设有一组权2,3,5,7,11,13,17,19,23,29,31,37,41。求相应的最优树。

实例

例　在通信中，设八进制数字出现的频率如下：

0：25%　　1：20%　　2：15%　　3：10%

4：10%　　5：10%　　6：5%　　7：5%

采用二元前缀码, 求传输数字最少的二元前缀码, 并求传输10^n(n≥2)个按上述比例出现的八进制数字需要多少个二进制数字？若用等长的（长为3）的码字传输需要多少个二进制数字?

解　用Huffman算法求以频率(乘以100)为权的最优2元树. 这里 w_1=5, w_2=5, w_3=10, w_4=10, w_5=10, w_6=15, w_7=20, w_8=25. 最优2元树如图所示.

实例（续）

编码： 0---01， 1---11， 2---001， 3---100

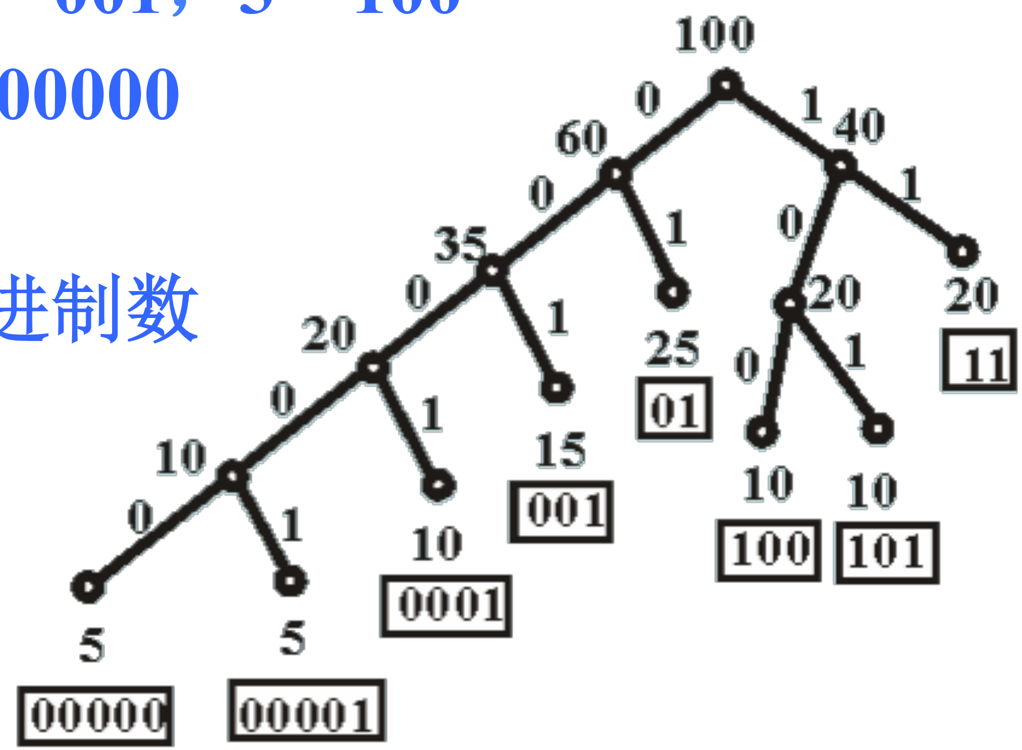
4---101， 5---0001， 6---00000

7---00001

传100个按比例出现的八进制数字所需二进制数字的个数为 $w(T)$=285.

传10^n(n≥2)个所用二进制数字的个数为2.85×10^n,

而用等长码(长为3)需要用 3×10^n个数字.



根树的遍历

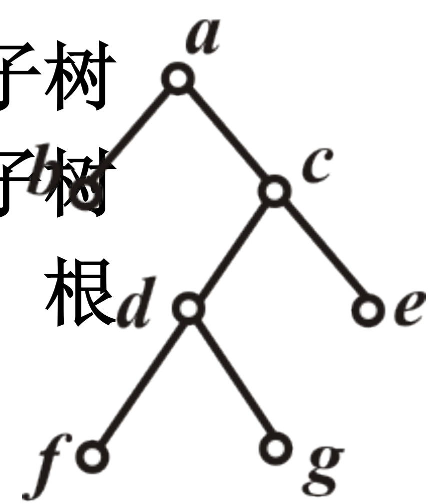
行遍(周游)根树T ： 对T 的每个顶点访问且仅访问一次.

行遍2元有序正则树的方式:

① 中序行遍法: 左子树、根、右子树

② 前序行遍法: 根、左子树、右子树

③ 后序行遍法: 左子树、右子树、根



例如, 对图所示根树按中序、前序、后序行遍法访问结果分别为:

$b\ \underline{a}\ (f\ \underline{d}\ g)\ \underline{c}\ e \quad \underline{a}\ b\ (\underline{c}\ (\underline{d}\ f\ g)\ e) \quad b\ ((f\ g\ \underline{d})\ e\ \underline{c})\ \underline{a}$

带下划线的是(子)树根, 一对括号内是一棵子树

算式的树形结构

用2元有序正则树表示算式：最高层次运算放在树根上,

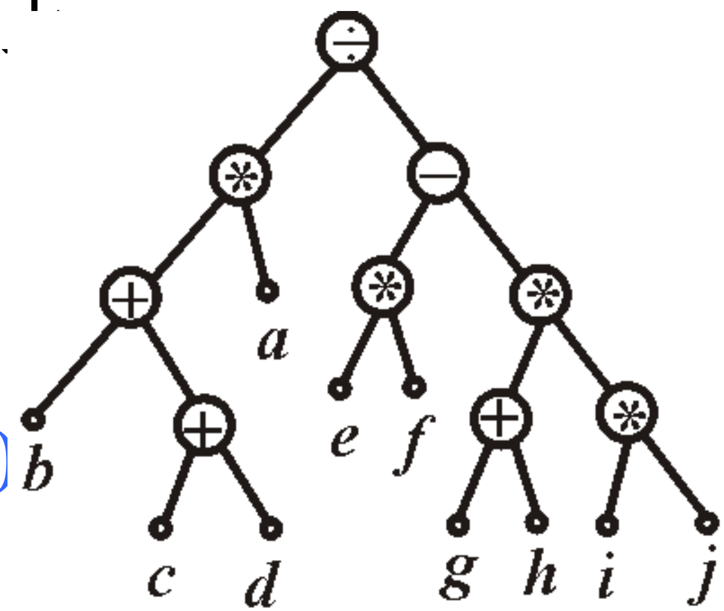
依次将运算符放在根子树的根

数放在树叶上, 规定被除数、

被减数放在左子树树叶上.

例如, 右图表示算式

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



算式的两种表示法

波兰符号法(前缀符号法)： 按前序行遍法访问表示算式的2元有序正则树，其结果不加括号，规定每个运算符号与其后面紧邻两个数进行运算.

逆波兰符号法(后缀符号法): 按后序行遍法访问，规定每个运算符与前面紧邻两数运算.

实例

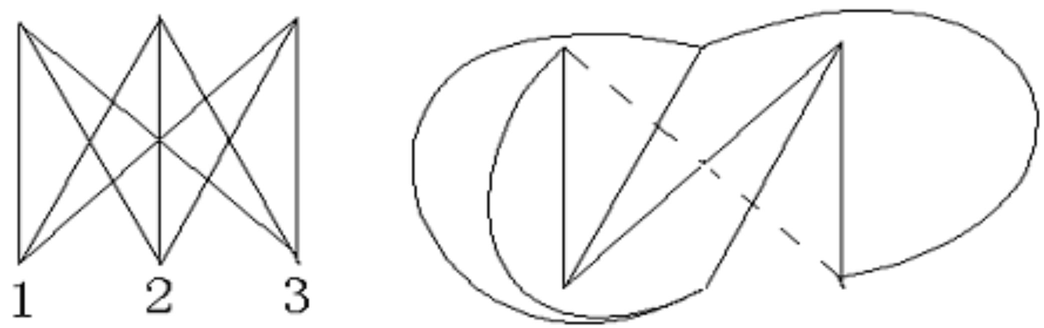
- 例：

前缀表达式 ↑-*3 3 * 4 2 5 的值是什么？

后缀表达式 3 2 * 2 ↑5 3 - 8 4/ * - 的值是什么？

平面图及图的着色

例：



有六个顶点的图如上，

试问：能否转变成与其等价的，但没有任何相交线的平面上的图？

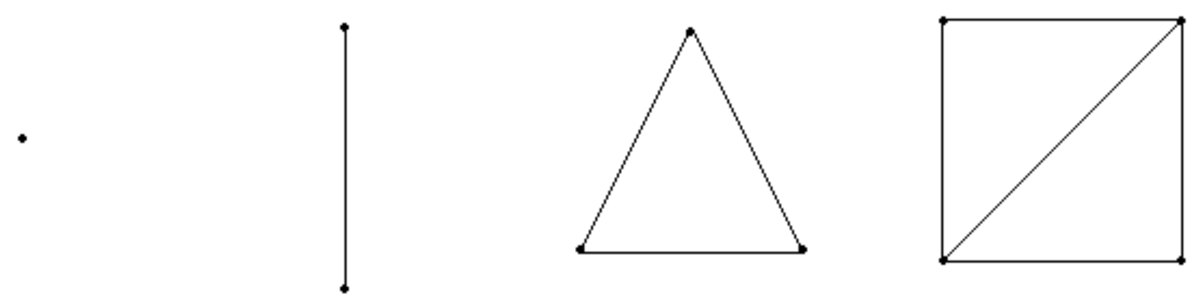
结论：不能

平面图及图的着色

定义　一个图G，如果能把它图示在一曲面S上，边与边只在顶点处相交，则称图G可嵌入曲面S。若图G可嵌入平面，则称为平面图。画出的无边相交的图称为G的平面嵌入，无平面嵌入的图称为非平面图。

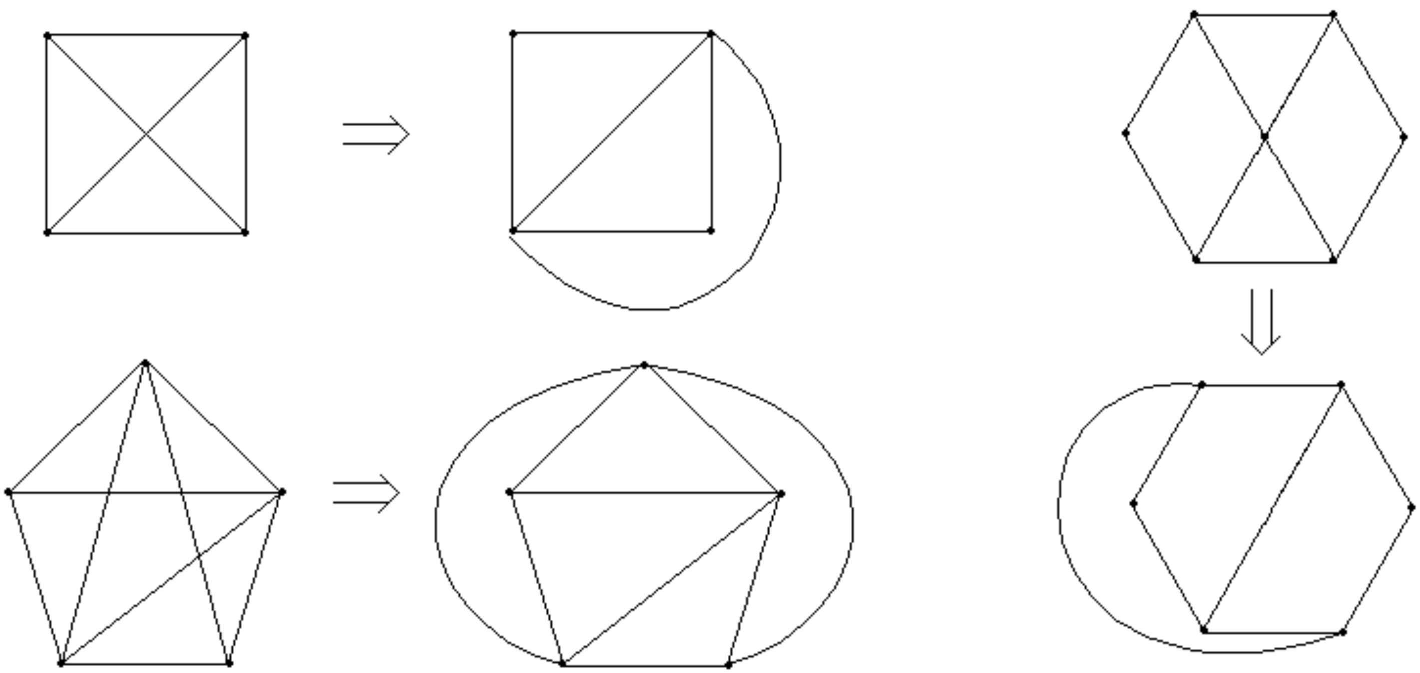
讨论定义：

(1)平面上的图，一开始就画成如定义所讲的图；

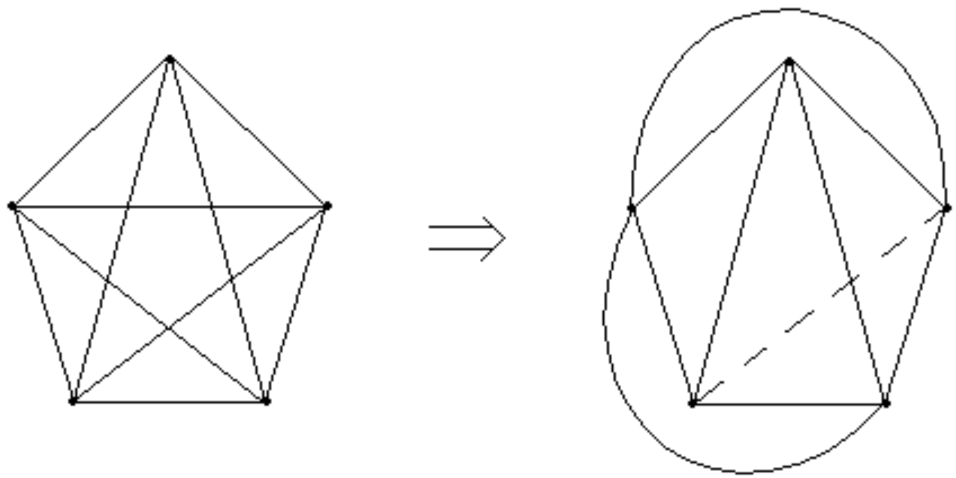


(2)原来在平面上的图形似交叉，

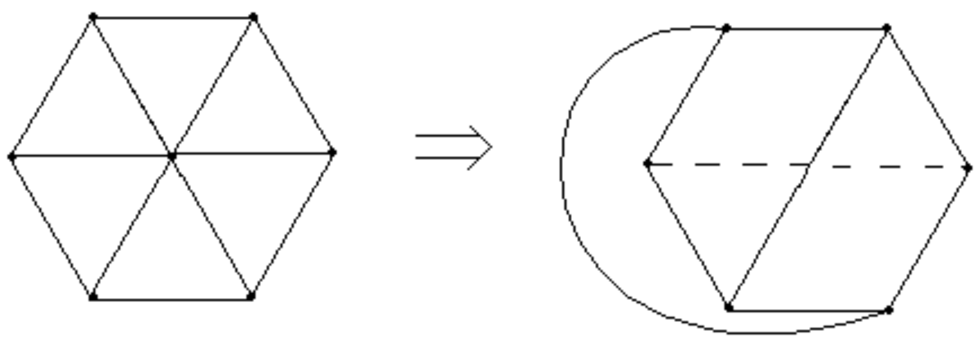
经过若干次的改画，变成符合定义所规定的图；



(3)下列图形均为非平面图



K_5 图



$K_{3,3}$ 图

平面图及图的着色

定理 若图G是平面图，则G的任何子图都是平面图。

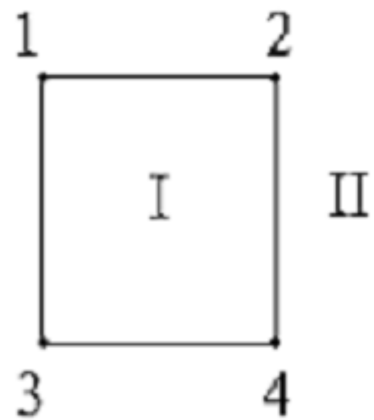
定理 若图G是非平面图，则G的任何母图都是非平面图。

推论 $K_n(n\geq 5)$和$K_{3,n}(n\geq 3)$都是非平面图。

定理 设G是平面图，则在G中加平行边或环后所得图还是平面图。

平面图及图的着色

定义　设G是平面图(且已经平面嵌入)，由G的边将G所在的平面划分成若干个区域，每个区域称为G的一个面。



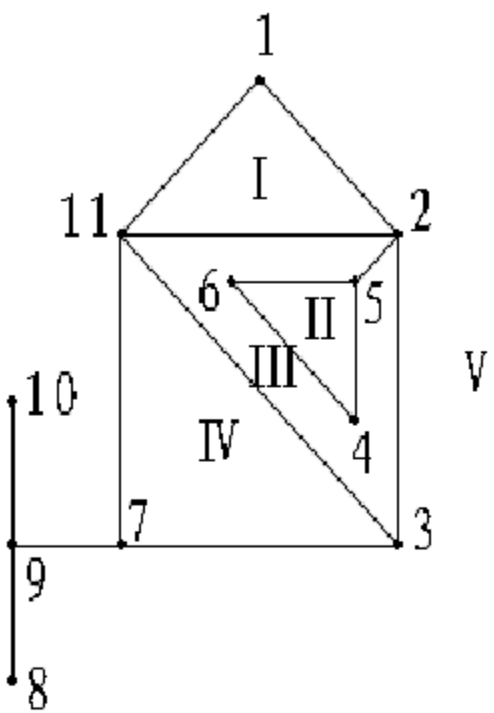
面　　I II

I 为有限面

II 为无限面

平面图及图的着色

例：



有五个面，其中：有限面：I、II、III、IV
无限面：V

平面图及图的着色

定理 设**G**是简单平面图，则**G**的最小度小于等于**5**。

- 对偶图。

设G为平面图，则G的对偶图为G*，且G*也为平面图。求图G的对偶图的方法如下：

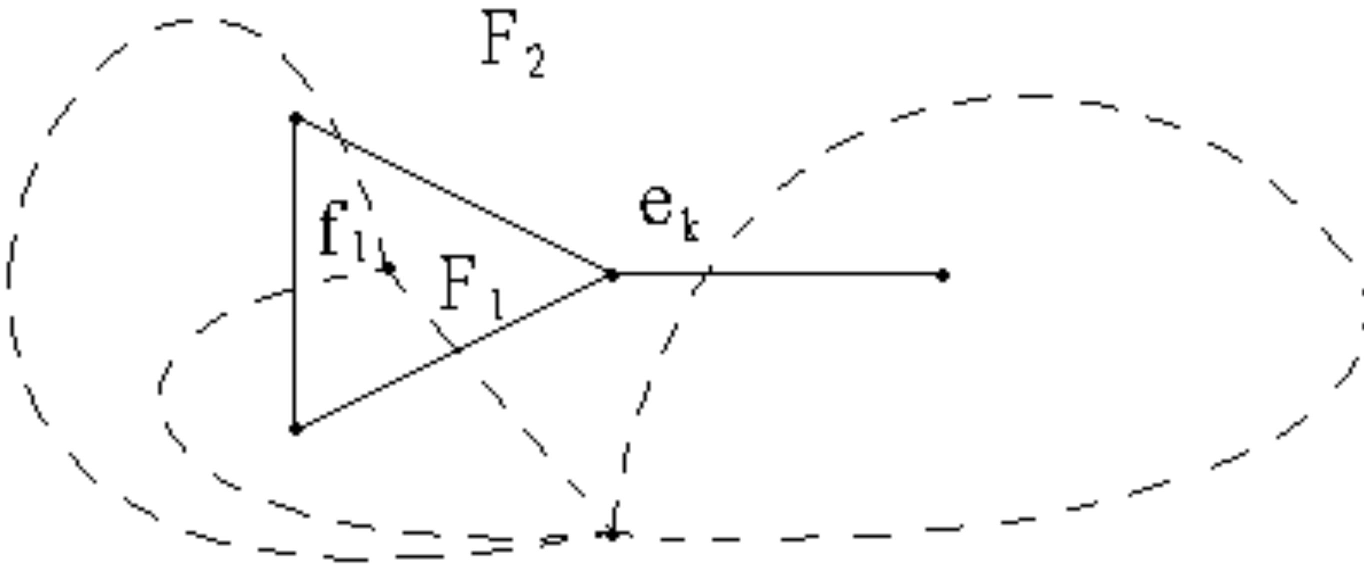
(1)将图G所有的面F_i(包括无限面)对应于G*的顶点f_i；

(2)对应二个相邻的面F_i ,F_j (即F_i ,F_j之间有公共边e_k)，对于每个公共边在f_i ,f_j之间作一条连线(即形成一条边(f_i,f_j))；

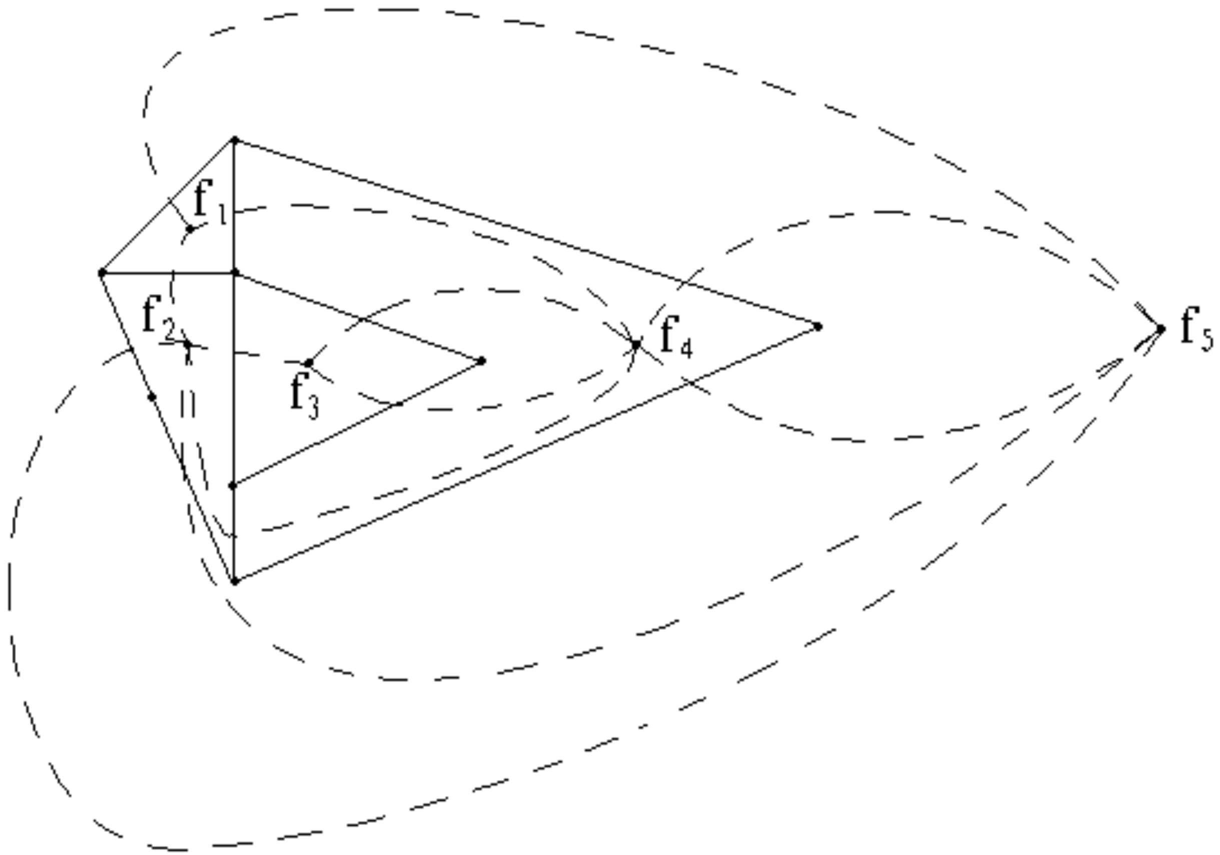
(3)当且仅当e_k只是F_i的边界时，f_i 恰存在一条自回路与e_k相交。

所得到的图即是图**G**的对偶图，记为**G***。

例：

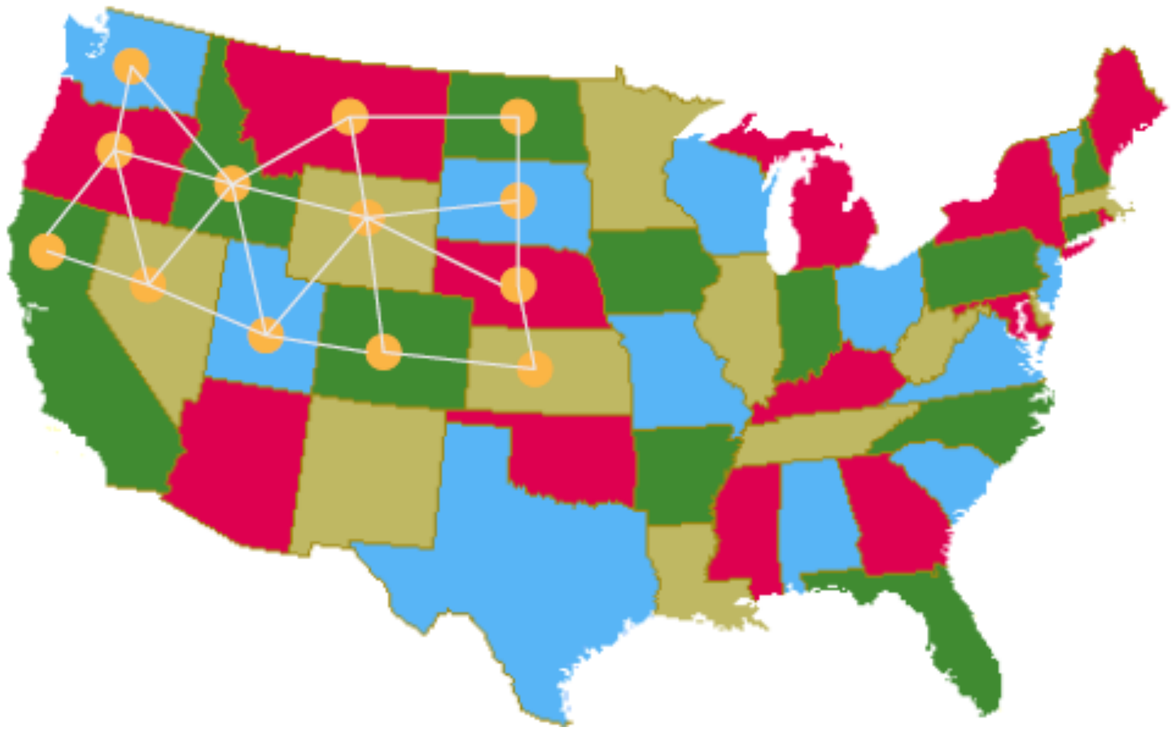


例：求G的对偶图



平面图及图的着色

❑平面图着色



平面图及图的着色

定义　给图**G**的正常着色(简称着色)是指对它每一个顶点指定一种颜色，使得没有二个相邻顶点有相同的颜色，若用了**n**种颜色着色，则称**G**为**n**色的，而对**G**用最少的颜色进行着色，称最少的颜色数为色数，记作**x(G)**。

平面图及图的着色

- 五色定理(**Heawood,1890**)：用**5种颜色**可以给任何简单连通平面图着色。

证明：对顶点数**v**用归纳法

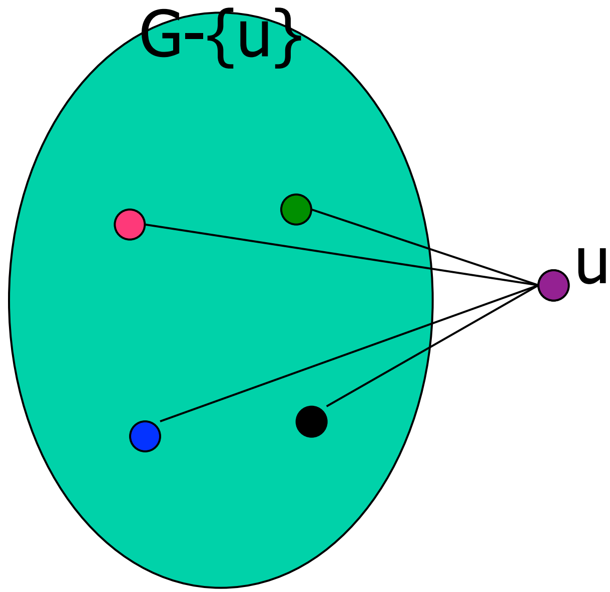
a)当**v**＝**1**，**2**，**3**，**4**，**5**时显然成立。

b)设**v**＝**k**时成立，现考察**v**＝**k+1**

已知必存在顶点**u**，使**deg(u)≤5**，在图**G**中删去**u**，得到**G-{u}**，由归纳假设知**G-{u}**可以用**5**种颜色着色。

平面图及图的着色

- 将u加入到G-{u}中，若deg(u)<5，必可对u正常着色，得到一个最多是五色的图G。

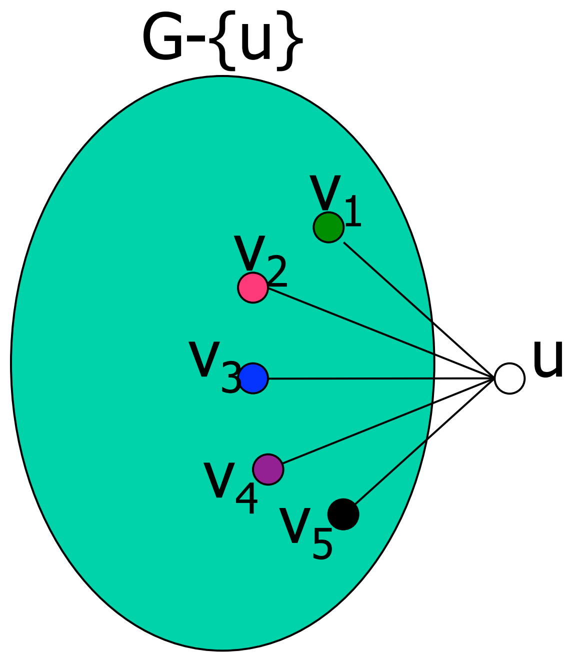


平面图及图的着色

- 将u加入到**G-{u}**中，若**deg(u)=5**。

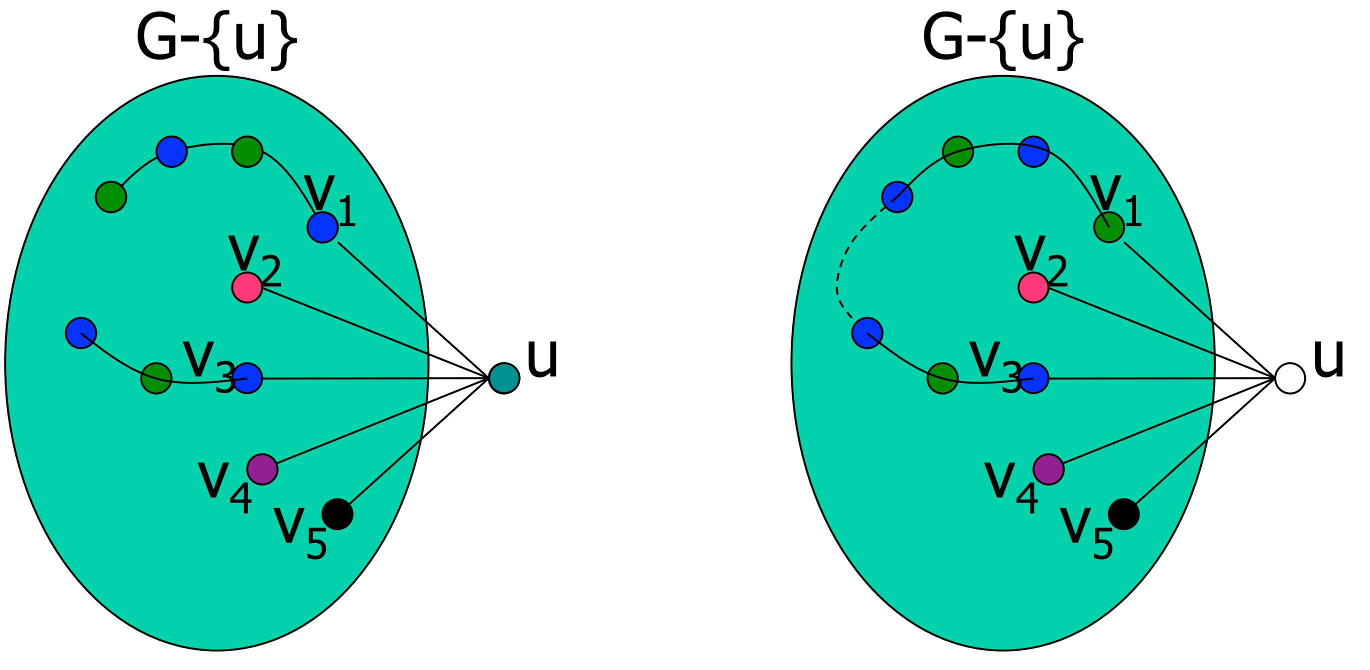
H为**G-{u}**中绿色和蓝色的顶点集合，

F为**G-{u}**中红色和紫色的顶点集合。



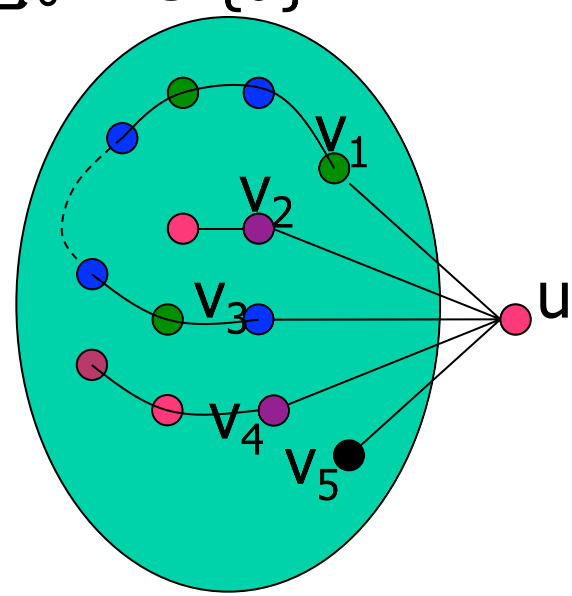
平面图及图的着色

- 若v_1与v_3属于顶点集H所导出子图的两个不同的连通分支中，将v_1所在分图中的蓝色和绿色对调，在u上着绿色。



平面图及图的着色

- 若v_1与v_3属于顶点集H所导出子图的同一个连通分支中，那么v_2与v_4将分别属于顶点集F所导出子图的两个不同连通分支中。在包含v_2的连通分支中将红色和紫色对调，对u着红色。 G-{u}



- 思考：K_n的色数是多少？