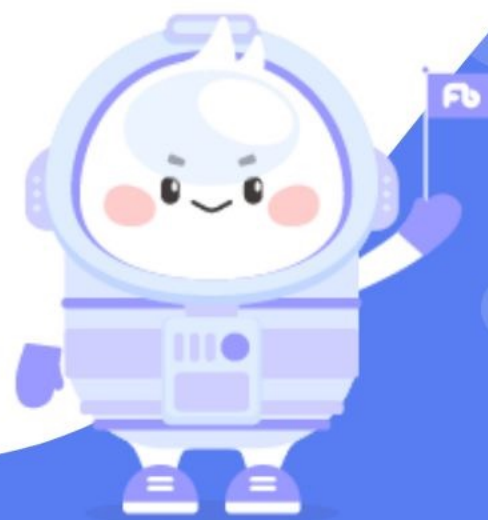


《信息技术》
数据结构与算法 4/5

► 讲师：孙珍珍

更多干货关注  粉笔教师教育  粉笔教师

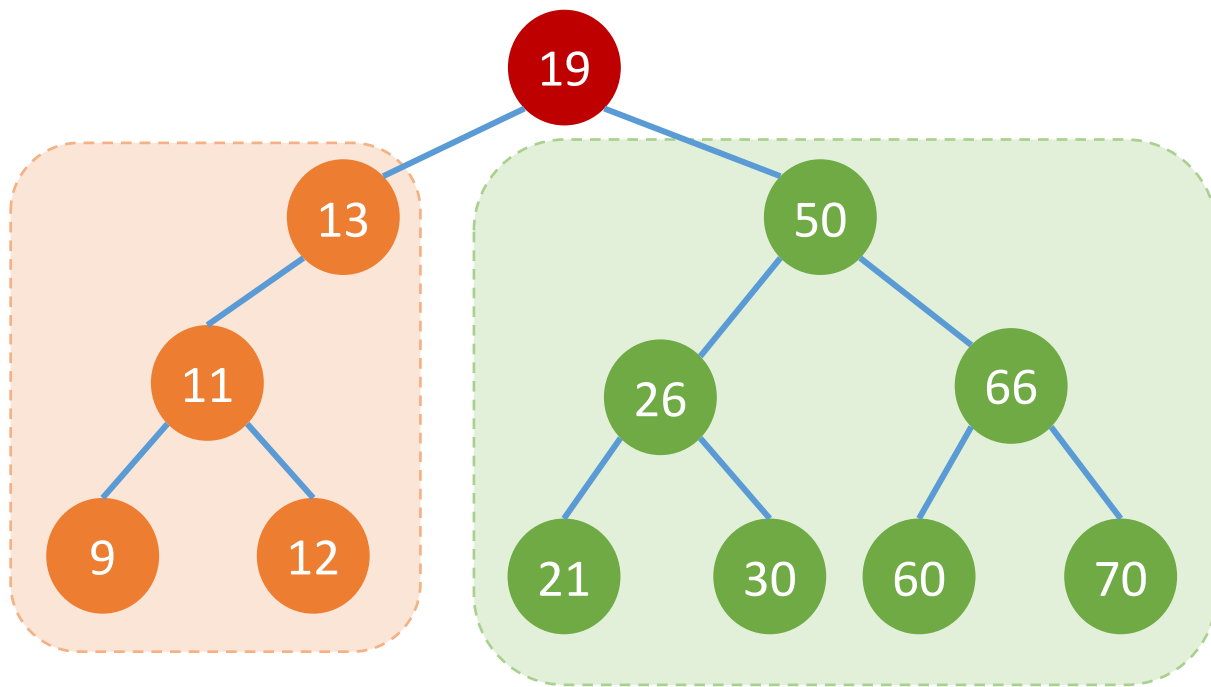


✿ 复习一下



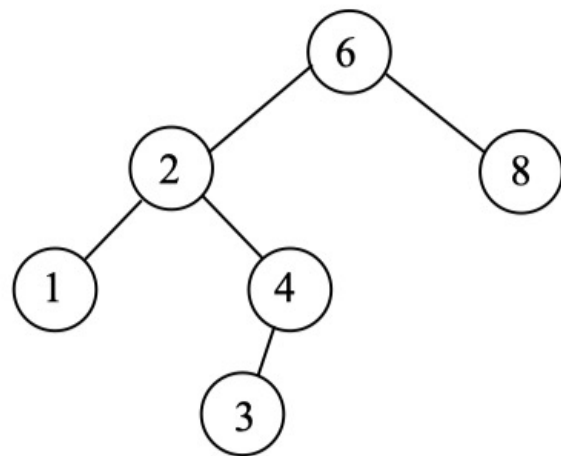
(一) 二叉排序树

1. 二叉排序树的定义

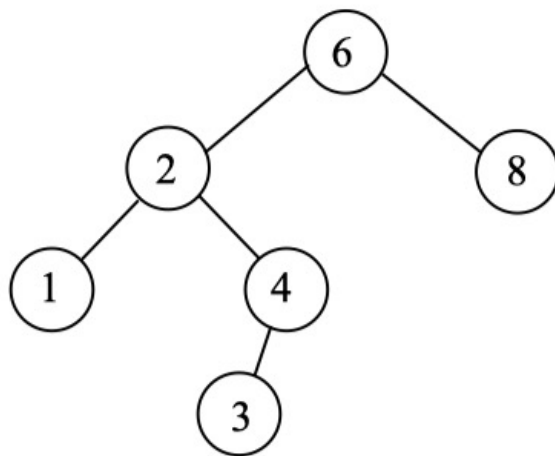


特点：左子树 < 根节点 < 右子树

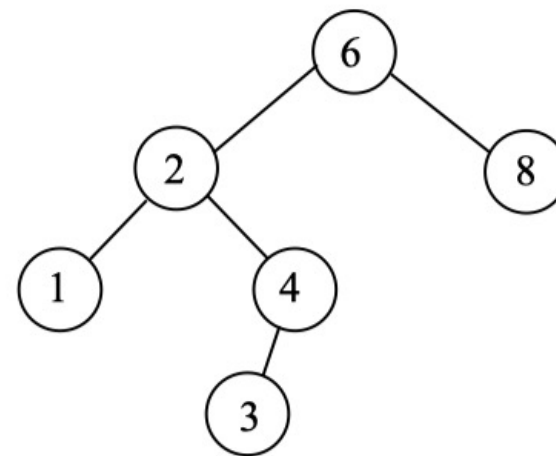
2~3. 二叉排序树的查找和插入



查找4



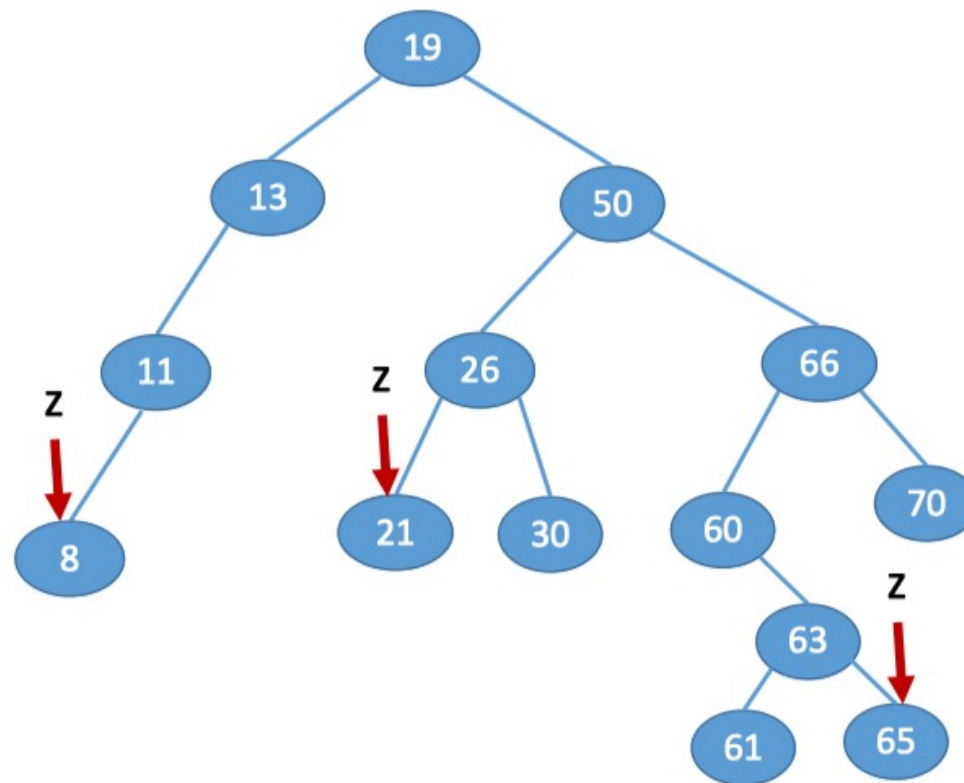
插入 7



插入 5

特点：从根节点开始，小在左，大在右

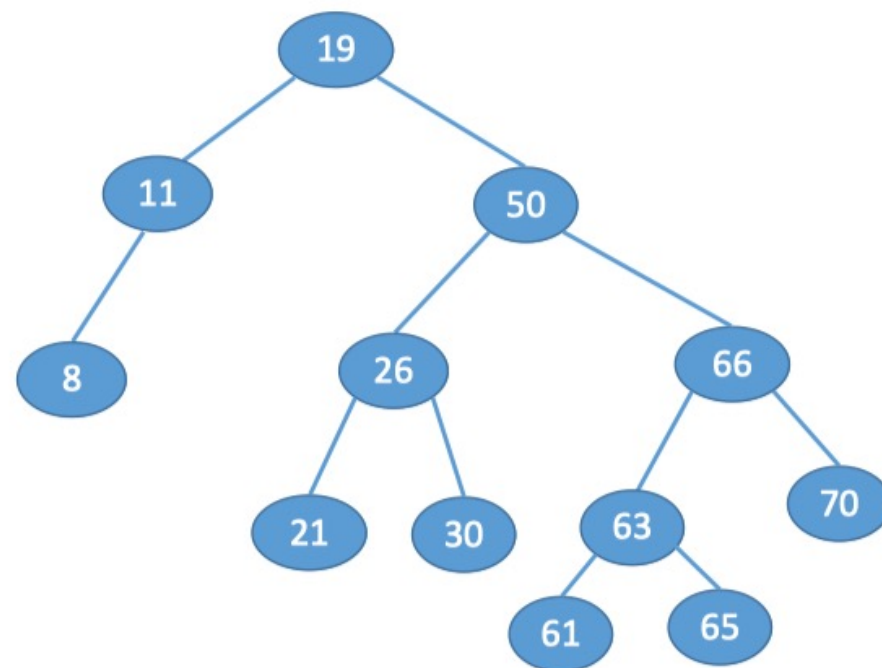
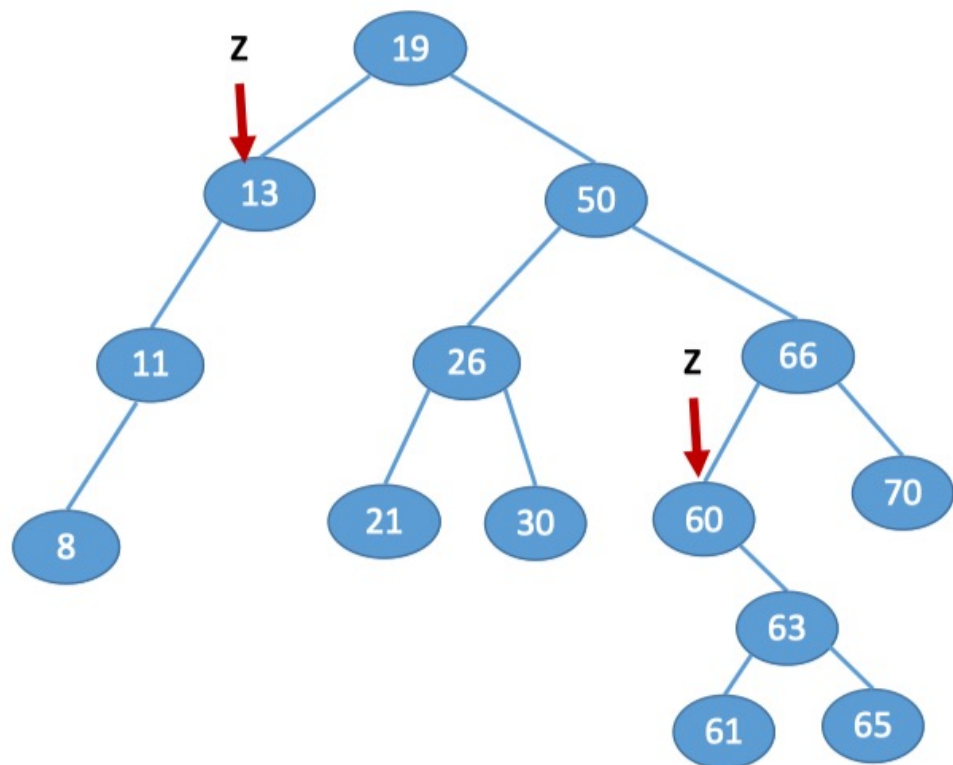
4. 二叉排序树的删除



总原则：左子树 < 根节点 < 右子树

情况1：z 是叶子节点，则直接删

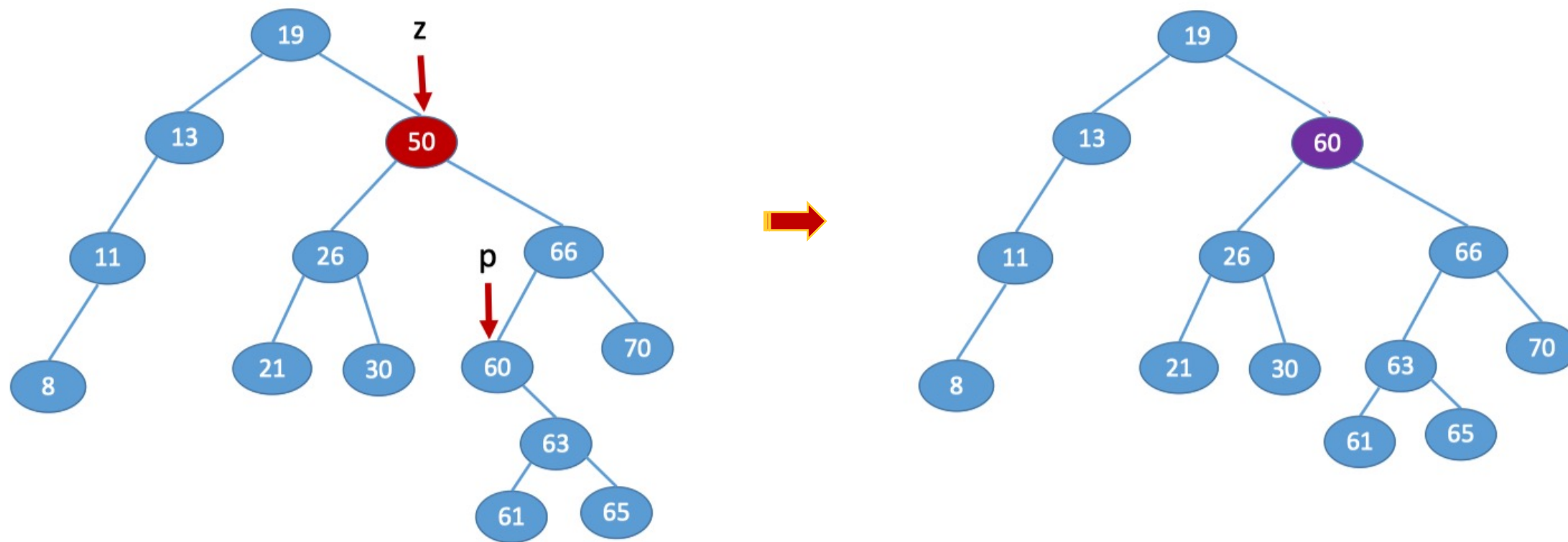
4. 二叉排序树的删除



总原则：左子树 < 根节点 < 右子树

情况2：z是单分支，则它的子树上移替代 z

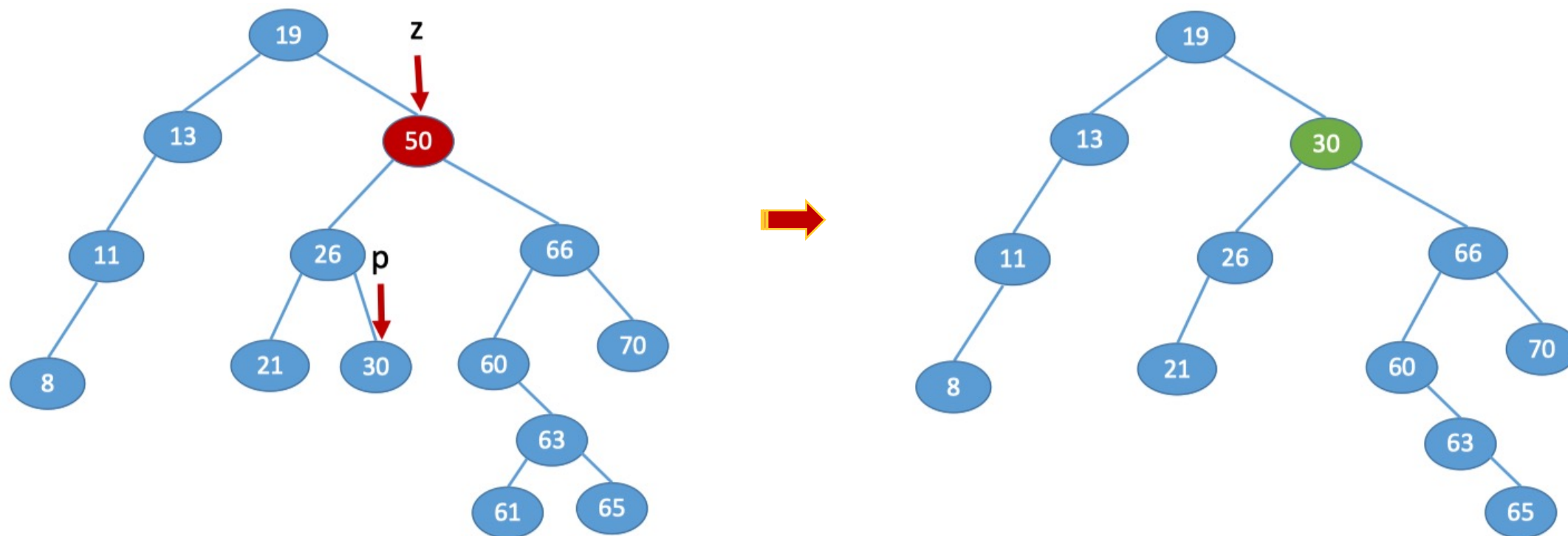
4. 二叉排序树的删除



总原则：左子树 < 根节点 < 右子树

情况3-1：z是双分支，则它的直接后继（右子树最小的值）代替它

4. 二叉排序树的删除



总原则：左子树 < 根节点 < 右子树

情况3-2：z是双分支，则它的直接前驱（左子树最大的值）代替它

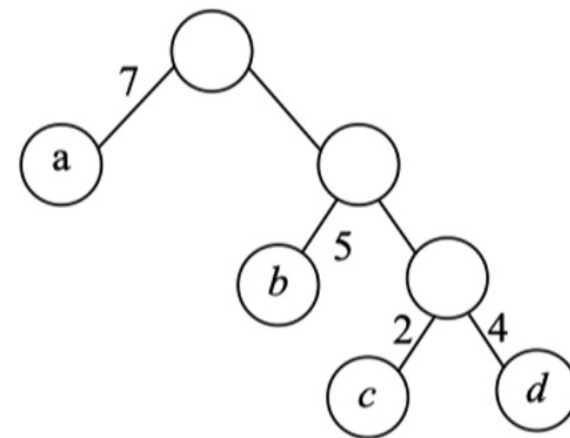
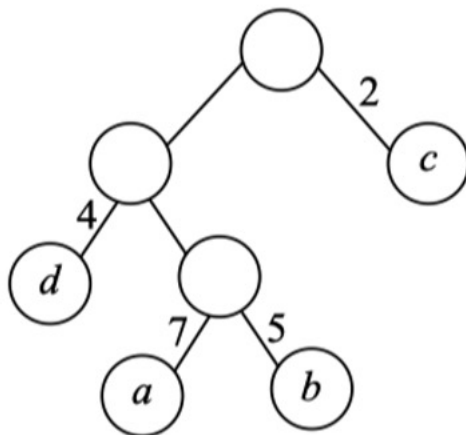
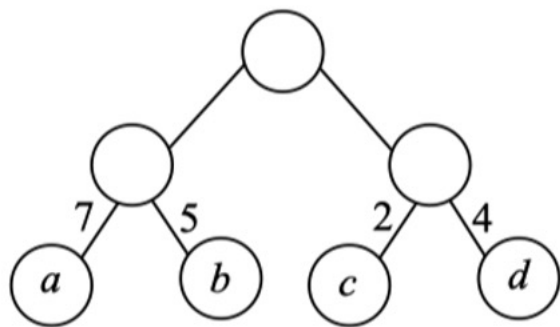
1. 哈夫曼树的定义

➤ 又称最优二叉树，是带权路径长度（WPL）最短的树。

✓ 权：为节点赋予某种意义的数值

✓ 节点的带权路径长度：从根到节点的路径长度与该节点上权值的乘积

✓ 树的带权路径长度：树中所有叶子节点的带权路径长度之和



2. 哈夫曼树的构造

【例】给定权值集 $w=\{7, 5, 2, 4\}$ ，构造关于 w 的哈夫曼树，并求其加权路径长度 WPL。

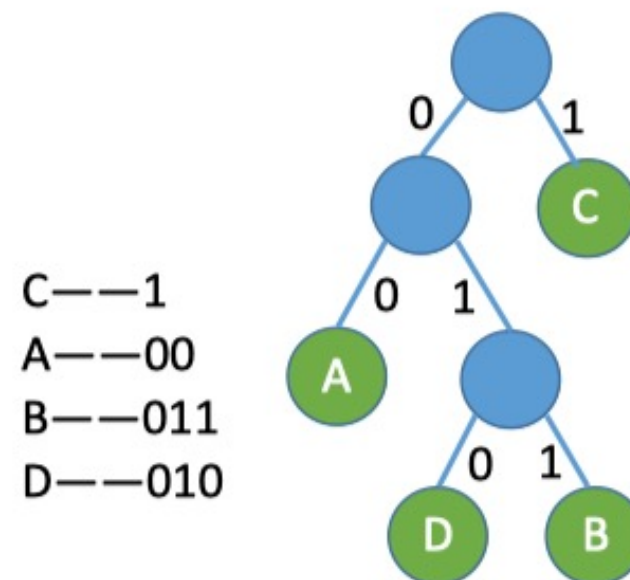
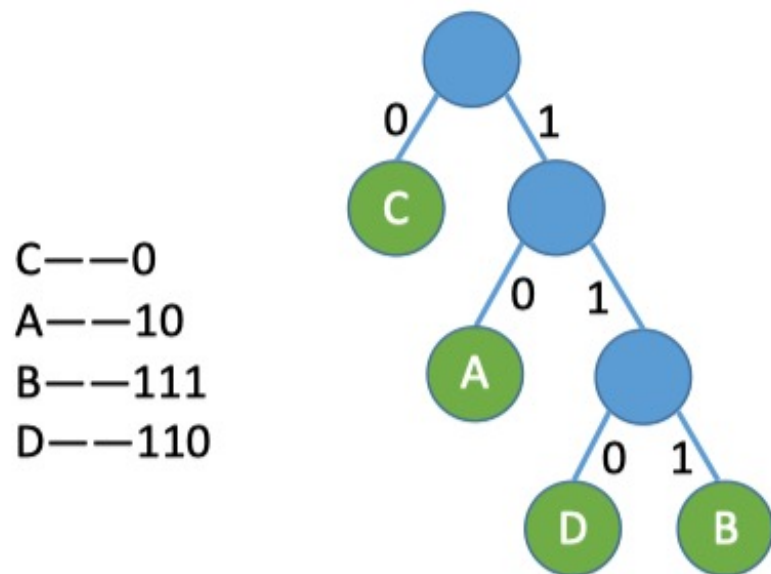
第1步：找最小的2个数

第2步：组成二叉树（左小右大）根为2数之和

第3步：删除这2个数，并加入根（2数之和）

第4步：重复上述步骤

3. 哈夫曼编码



编码规则：左分支赋予0，右分支赋予1

已知字符集{a , b , c , d , e , f} , 若各字符出现的次数分别为6、 3、 8、 2、 10、 4 , 则对应字符集中各字符的哈弗曼编码可能是 ()。

- A. 00, 1011 , 01 , 1010, 11, 100
- B. 00, 100, 110,000, 0010, 01
- C. 10 , 1011 , 11 , 0011 , 00, 010
- D. 0011 , 10, 11 , 0010, 01 , 000

第1步：构造哈夫曼树

第2步：左0右1得到编码



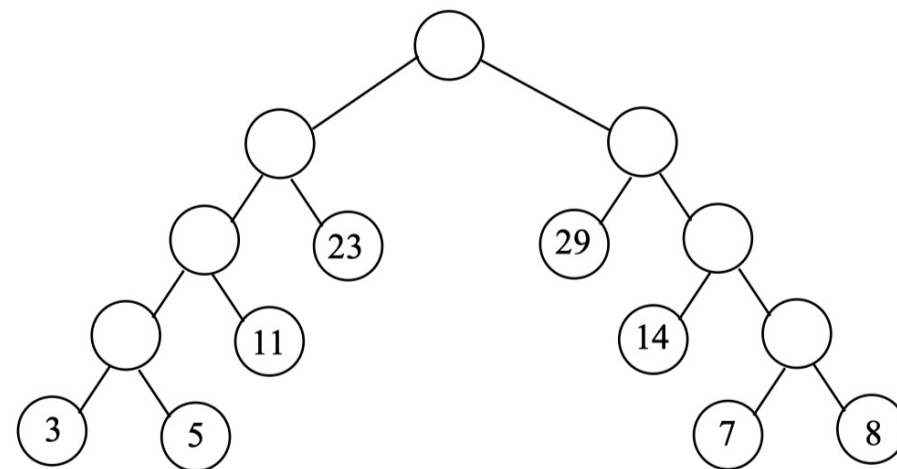
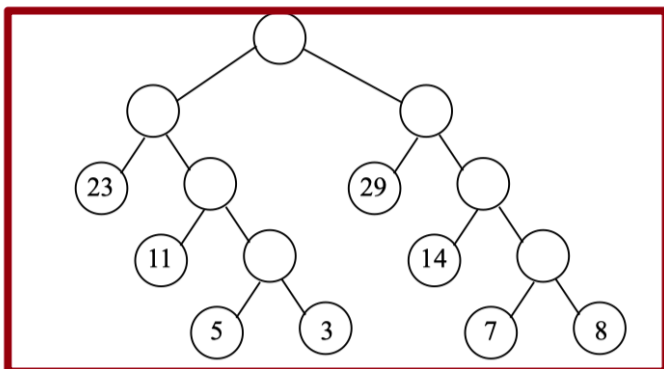
【例】已知某系统在通信联络中只可能出现8种字符，其概率分别为0.05，0.29，0.07，0.08，0.14，0.23，0.03，0.11，试设计哈夫曼编码。

第1步：构造哈夫曼树

第2步：左0右1得到编码

【例】已知某系统在通信联络中只可能出现 8 种字符，其概率分别为 0.05, 0.29, 0.07, 0.08, 0.14, 0.23, 0.03, 0.11, 试设计哈夫曼编码。

第一步：根据其出现的概率可设 8 个字符的权值为： $w = (5, 29, 7, 8, 14, 23, 3, 11)$ ，构造哈夫曼树如下图所示。



第二步：将树的左分支标记为 0，右分支标记为 1，便得到其哈夫曼编码表如下所示。

HC	
1	0 1 1 0
2	1 0
3	1 1 1 0
4	1 1 1 1
5	1 1 0
6	0 0
7	0 1 1 1
8	0 1 0

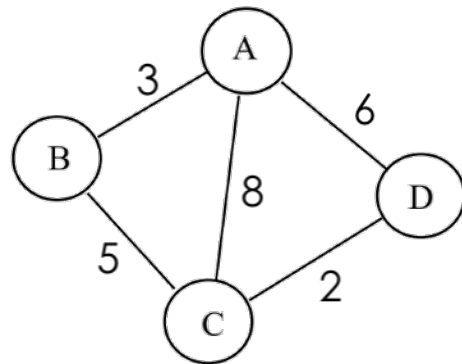
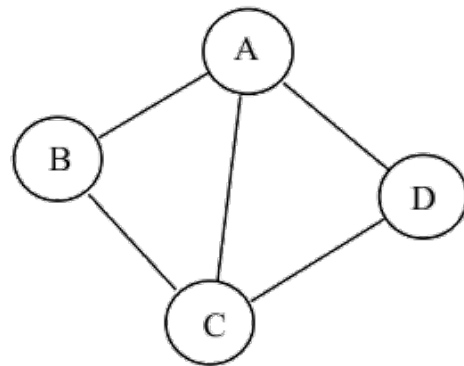
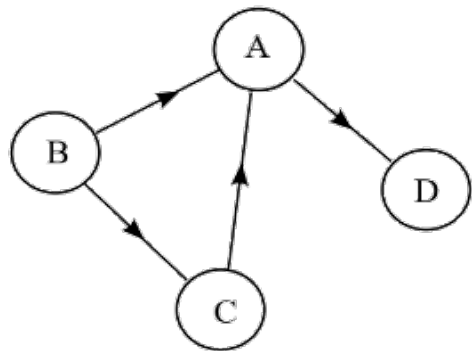


HC	
1	0 0 0 1
2	1 0
3	1 1 1 0
4	1 1 1 1
5	1 1 0
6	0 1
7	0 0 0 0
8	0 0 1

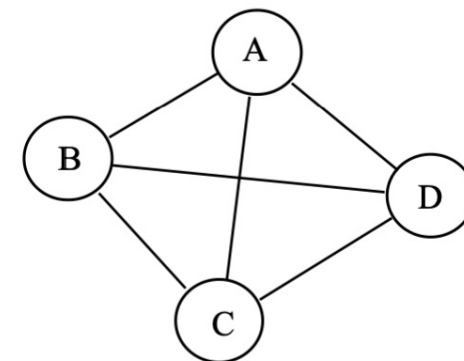
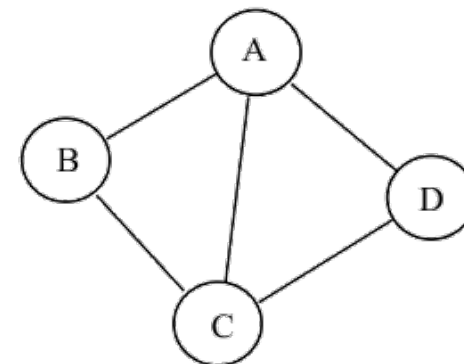
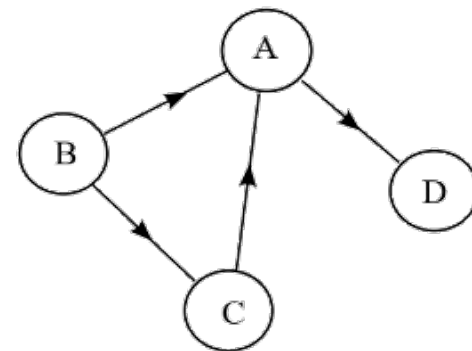


第五节 图

术语	说明
1.有向图	边上有箭头。如：AC弧表示为 $\langle C, A \rangle$
2.无向图	边上无箭头。如：AC边表示为 (A, C) 或 (C, A)
3.完全图	任意两顶点都有边/弧 分：有向完全图、无向完全图
4.子图	某个图的组成部分
5.邻接、依附、关联	两个顶点之间有边/弧 两个顶点互为邻接点、边依附顶点、边和顶点相关联
6.度	与顶点相关联的边/弧的个数 分：度、入度、出度
7.路径	从顶点A到顶点B，依次经过的顶点序列
8~9.连通	任意两个顶点之间都有通路 分：连通图、强连通图
10.权/网	边上有数值



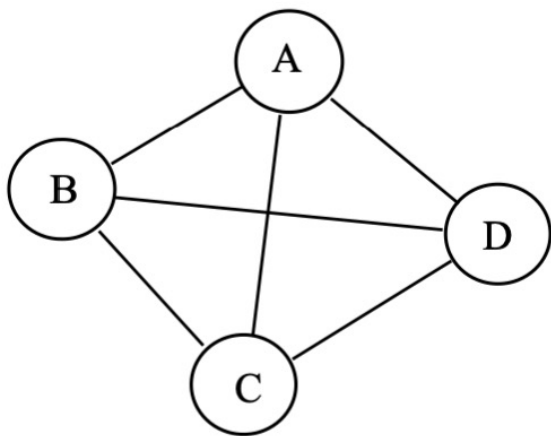
术语	说明
1.有向图	边上有箭头。如：AC弧表示为 $\langle C, A \rangle$
2.无向图	边上无箭头。如：AC边表示为 (A, C) 或 (C, A)
3.完全图	任意两顶点都有边/弧 分：有向完全图、无向完全图
4.子图	某个图的组成部分
5.邻接、依附、关联	两个顶点之间有边/弧 两个顶点互为邻接点、边依附顶点、边和顶点相关联
6.度	与顶点相关联的边/弧的个数 分：度、入度、出度
7.路径	从顶点A到顶点B，依次经过的顶点序列
8~9.连通	任意两个顶点之间都有通路 分：连通图、强连通图
10.权/网	边上有数值



3. 完全图

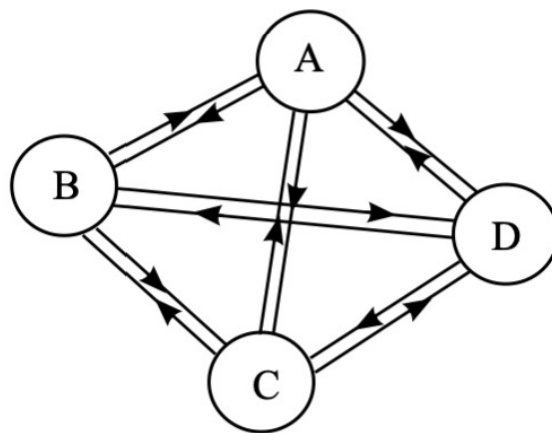
(1) 完全无向图

✓ 特点：有 $n(n-1)/2$ 条边



(2) 完全有向图

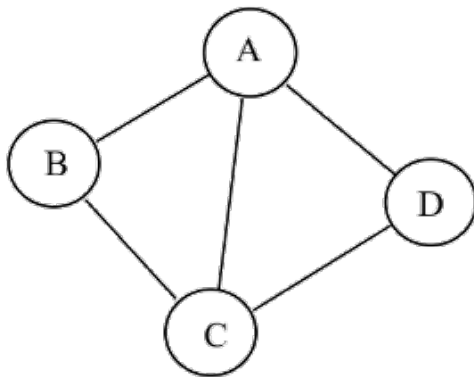
✓ 特点：有 $n(n-1)$ 条边



6. 度、入度、出度

➤ 度 (TD)

✓ 和顶点 v 相**关联**的边的数目



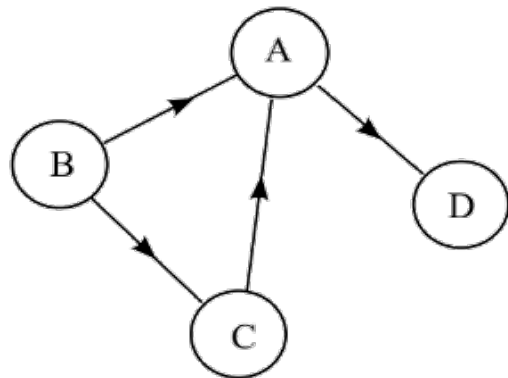
$$\sum_{i=1}^n TD(v_i) = 2e$$

➤ 入度 (ID)

✓ 以顶点 v 为**头**的弧的数目

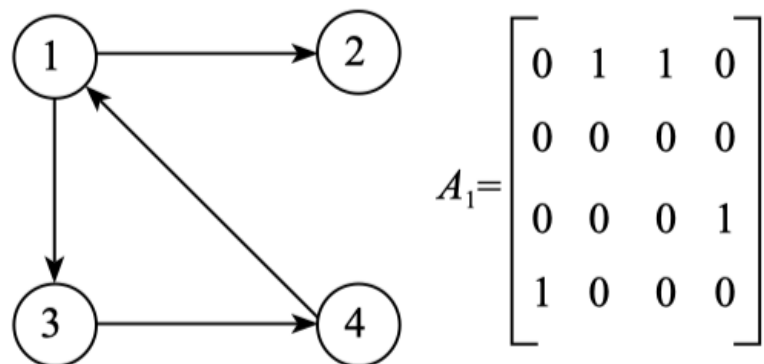
➤ 出度 (OD)

✓ 以顶点 v 为**尾**的弧的数目

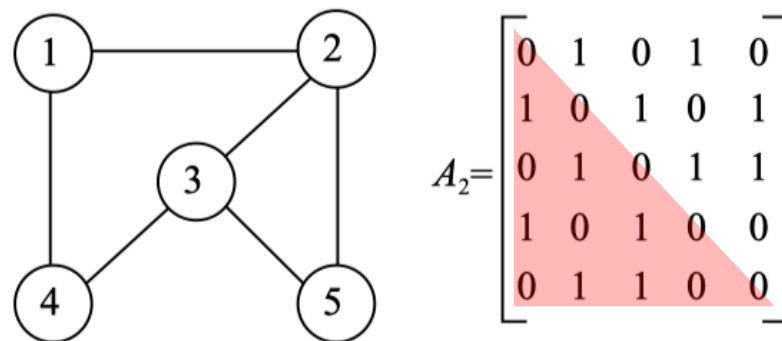


$$\sum_{i=1}^n ID(v_i) = \sum_{i=1}^n OD(v_i) = e$$

(一) 邻接矩阵

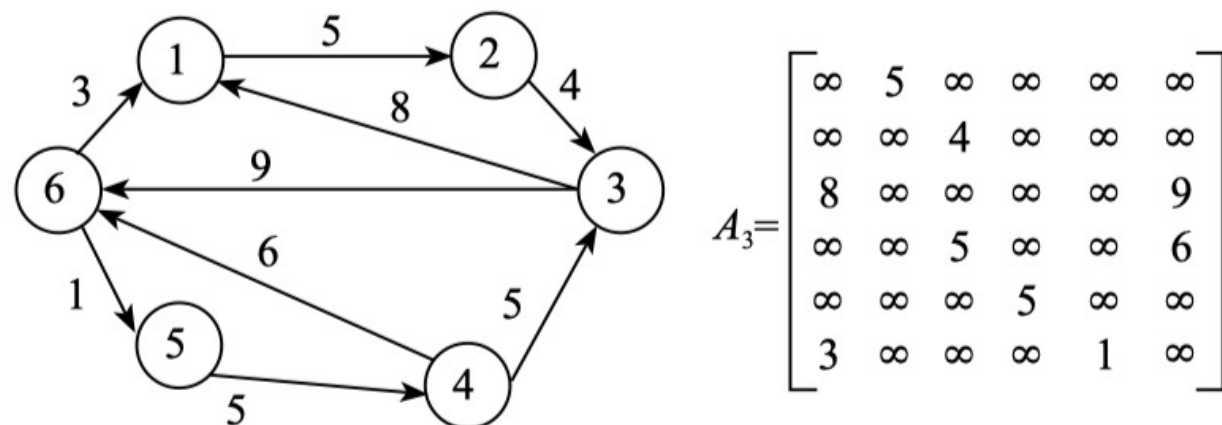


(a) 有向图 G_1 及其邻接矩阵



(b) 无向图 G_2 及其邻接矩阵

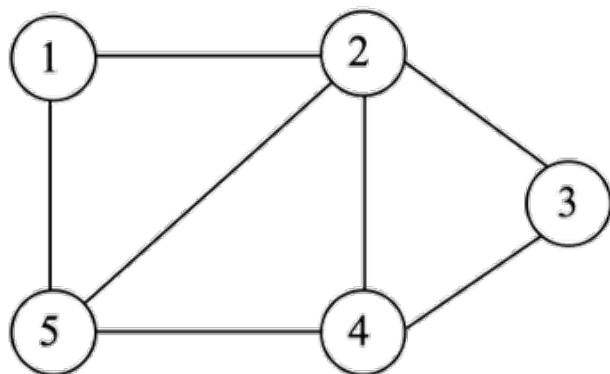
图：有边写1，无边写0



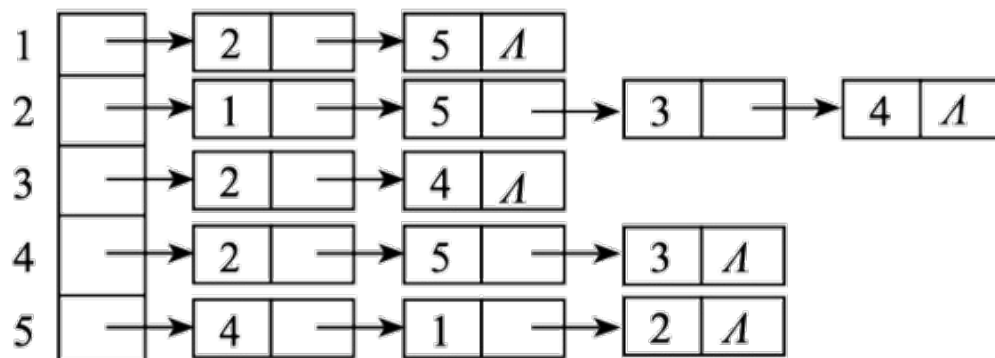
(c) 网及其邻接矩阵

网：有边写值，无边写 ∞

(二) 邻接表

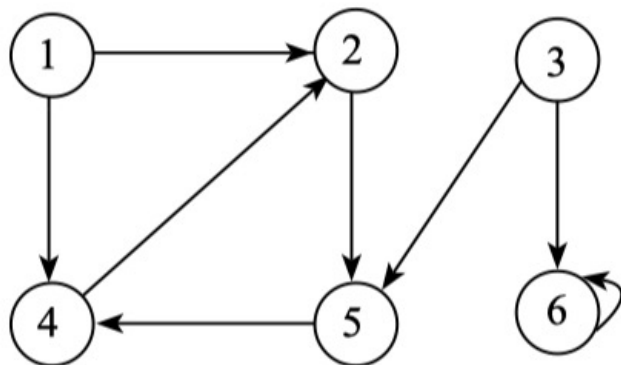


(a) 无向图 G

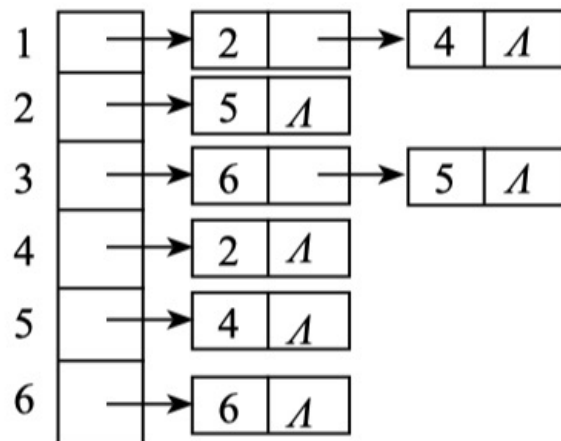


(b) 无向图 G 的邻接表的表示

无向图：写与该顶点相连的



(c) 有向图 G

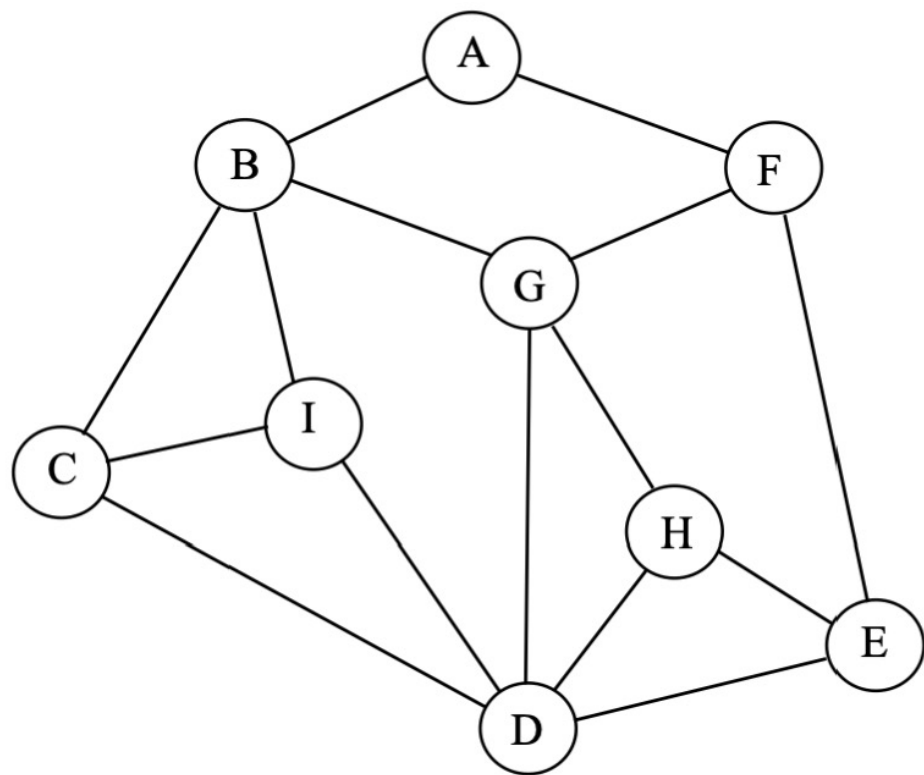


(d) 有向图 G 的邻接表的表示

有向图：写由该顶点出发的

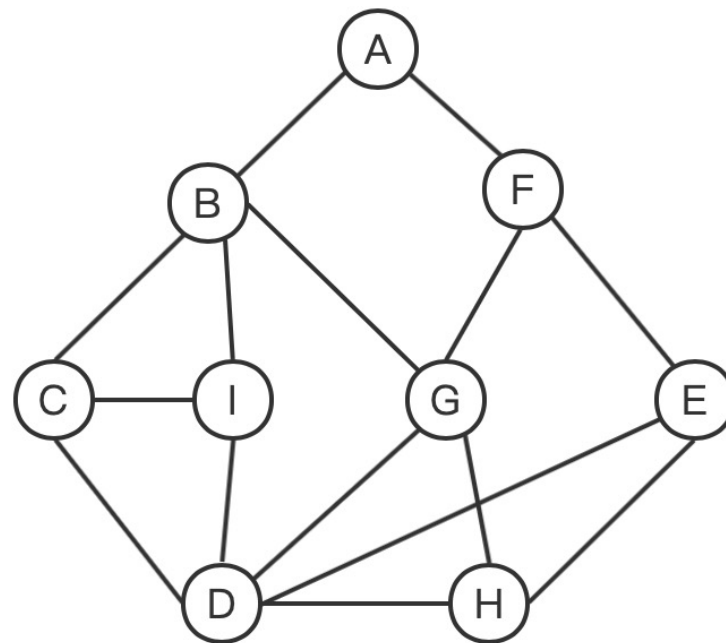
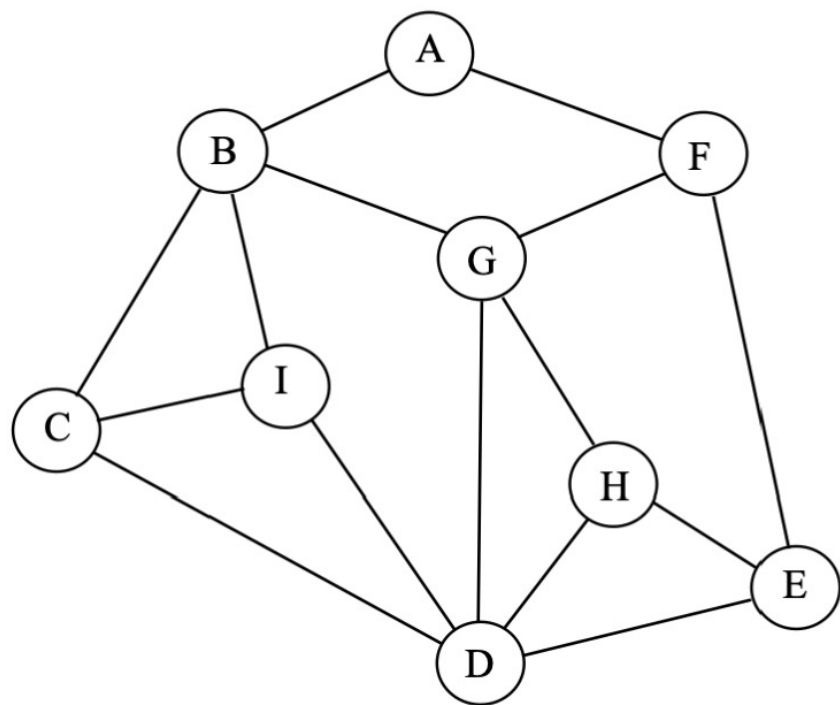
(一) 深度优先遍历

➤ 类似树的先序遍历，从任意顶点出发，沿一个方向遍历相邻顶点，然后回溯



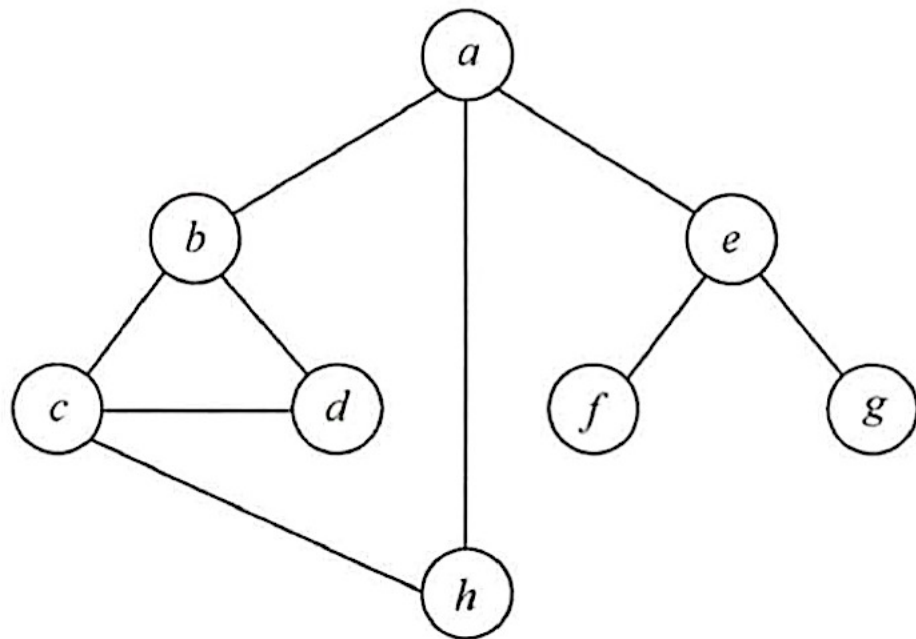
(二) 广度优先遍历

➤ 类似树的层遍历，从任意顶点出发，先遍历完该顶点所有相邻顶点，再遍历下一层



若对如下无向图进行遍历，则下列选项中，不是广度优先遍历序列的是（ ）。

- A. h、c、a、b、d、e、g、f
- B. e、a、f、g、b、h、c、d
- C. d、b、c、a、h、e、f、g
- D. a、b、c、d、h、e、f、g



◆旅行商问题（TSP）

- 从一个城市出发，需要经过所有城市一次并且仅一次之后，回到出发城市。
- 要求总路程最短。

◆两种解决方法（贪心）

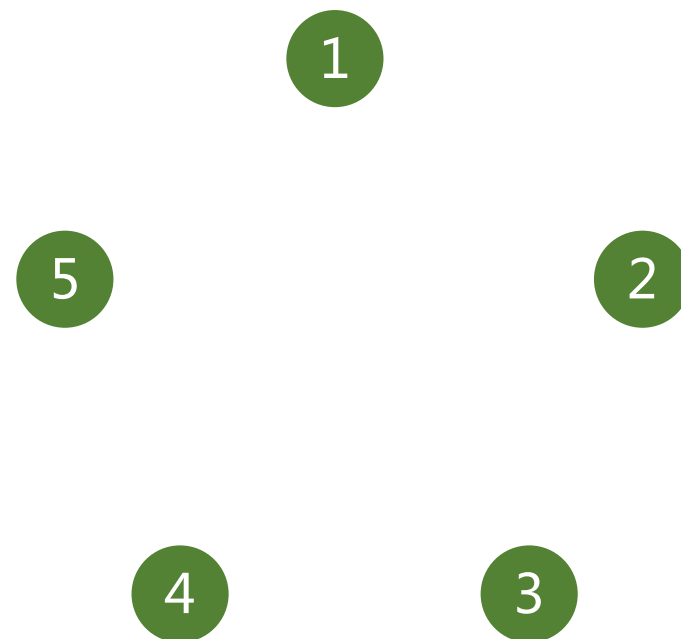
- 最短链接策略【关注的是边】
- 最近邻点策略【关注的是顶点】

【例】假设 5 个相互可直达且距离已知的城市如下图 (a) 所示，采用最短链接策略从城市 1 出发回到城市 1。

$$C = \begin{bmatrix} \infty & 3 & 3 & 2 & 6 \\ 3 & \infty & 7 & 3 & 2 \\ 3 & 7 & \infty & 2 & 5 \\ 2 & 3 & 2 & \infty & 3 \\ 6 & 2 & 5 & 3 & \infty \end{bmatrix}$$

(a) 5 城市的代价矩阵

城市	距离	可用否



第1步：按距离升序排列

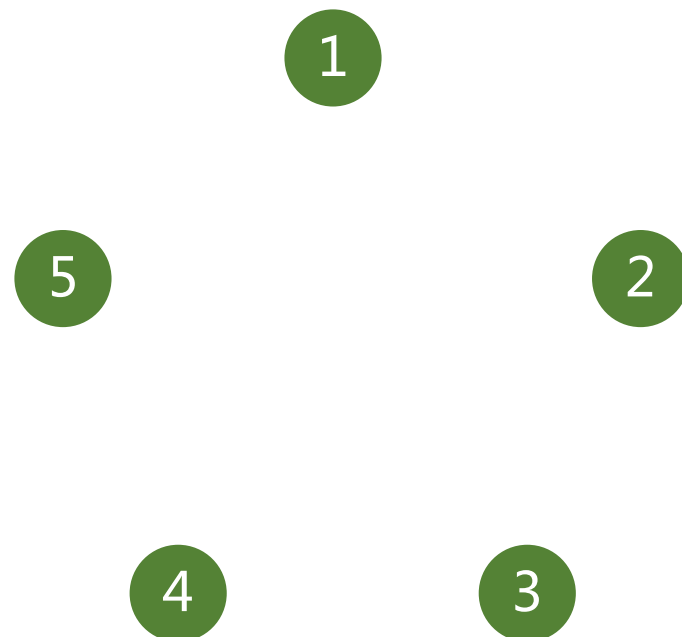
第2步：依次加入最短边

- 要求①：无回路
- 要求②：每个顶点 ≤ 2 条边

【例】假设 5 个相互可直达且距离已知的城市如下图 (a) 所示，采用最近邻点策略从城市 1 出发回到城市 1。

$$C = \begin{bmatrix} \infty & 3 & 3 & 2 & 6 \\ 3 & \infty & 7 & 3 & 2 \\ 3 & 7 & \infty & 2 & 5 \\ 2 & 3 & 2 & \infty & 3 \\ 6 & 2 & 5 & 3 & \infty \end{bmatrix}$$

(a) 5 城市的代价矩阵



第1步：找顶点A相邻的最短路径，确定下一个顶点B

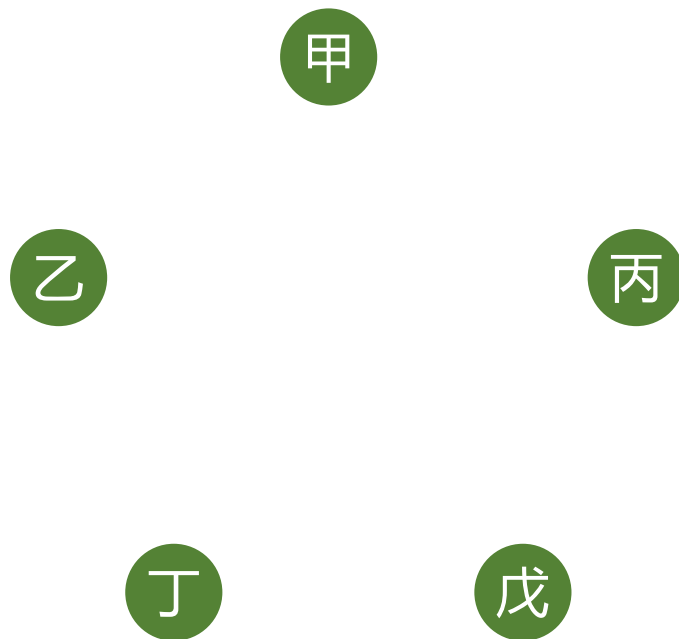
第2步：从B出发，继续找B相邻的最短路径

- 要求①：无回路
- 要求②：选未到过的顶点

(2021 下·高中) “绿水青山就是金山银山”，加强环境保护对经济社会发展十分必要，长江和重要支流进入十年禁渔期，一大批长江保护巡查员常年在“水上漂”，目的就是及时发现不法分子对长江水域的各种破坏活动。有五个相互可直达且距离已知的工作站甲、乙、丙、丁、戊，如下表所示，某巡查员从工作地甲出发，其他四个工作站巡查完后再回到工作站甲。产生式得出最短巡查路线的距离 x 是()。

- A. 42
B. 38
C. 36
D. 47

	甲	乙	丙	丁	戊
甲	—	7	5	10	10
乙	7	—	7	10	10
丙	5	7	—	6	9
丁	10	10	6	—	8
戊	10	10	9	8	—





术语

无向图 — 有向图 — 邻接



度

无向图-度 — $\Sigma \text{度} = 2 \times \Sigma \text{边}$

有向图-入度和出度 — $\Sigma \text{入度} = \Sigma \text{出度} = \Sigma \text{边}$

会求



完全图

完全无向图 — 任意顶点间均有边 — 有 $n(n-1)/2$ 条边

完全有向图 — 任意顶点间均有方向相反的弧 — 有 $n(n-1)$ 条弧

1. 会认图
2. 会求

连通图 — 任意顶点间都有通路

会认图

存储

邻接矩阵 — 有边写1, 无边写0 (图) — 有边写权值, 无边写 ∞ (网)

邻接表 — 写与该顶点相连的 (无向图) — 写从该顶点出发的 (有向图)

会认



遍历

深度优先 — = 树的先序遍历 — 找一个邻接节点, 一路向下, 然后再回溯

广度优先 — = 树的层遍历 — 找所有邻接节点, 然后再下一层



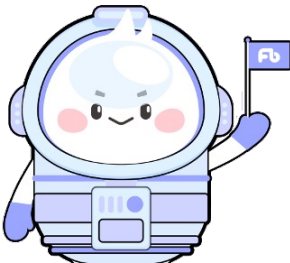


有疑问没？等你吖

下节内容

第六节	数据查找算法	320
第七节	数据排序算法	321
第八节	程序基础	332

P320 ~ P335



岸上等你

THE TEST

光芒万丈
不负理想

粉笔
教师



机读卡

姓名:

考号

