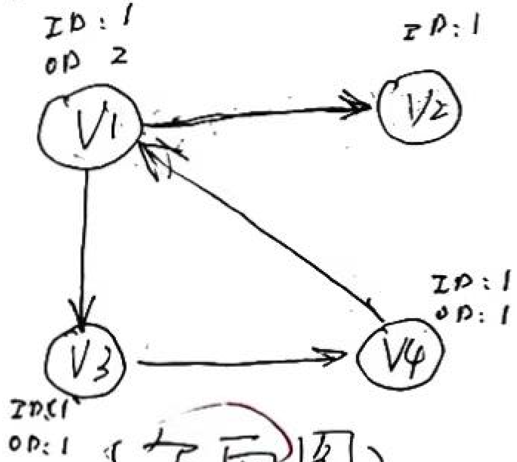
图

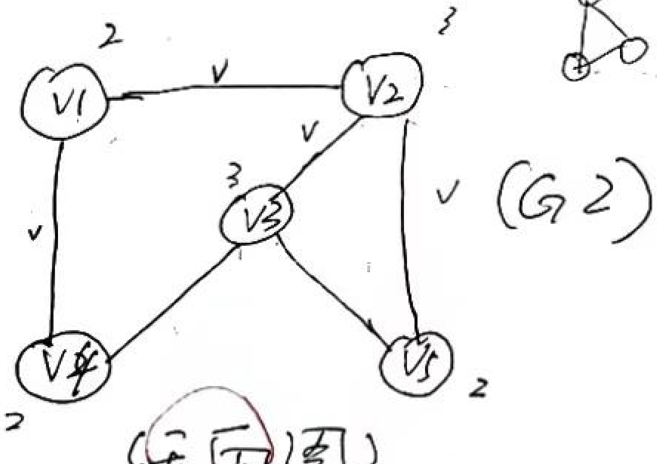
1. 图是一种数据结构。（逻辑结构）

2. 图例：

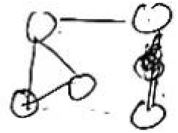


(G1)

（有向图）



（无向图）



3. 术语：

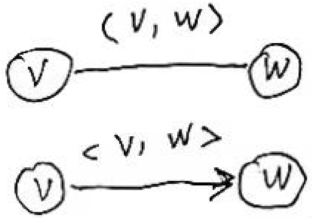
① 顶点——图中的数据元素。

② 弧（边）（Arc）：

若 <v, w> ∈ VR，则称为从顶点v到顶点w的一条弧（边）

（V：表示顶点的有穷非空集合 → 顶点的集合

（VR：两个顶点之间边的集合 ）



弧尾 弧头.
初始点。 终端点.

无向图：

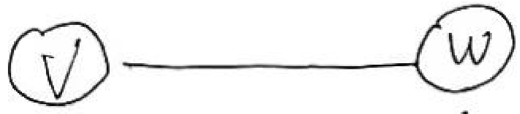
若有 $(v,w) \in VR$，就有 (w,v) 属于 VR，

则 VR 是对称的

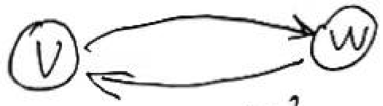
$\Rightarrow$ $\underline{(v,w)}$

边(Edge)——以无序对 (v,w) [代替两个]

有序对 $\langle v,w\rangle$, $\langle w,v\rangle$



邻接点 邻接点



<V,W>

<W,V>

V是W的入边邻接点.

W是V的出边邻接点.

图的表示：二元组表示法

$<v, w>$

A> 表示有向图

出边：

$G1 = (V, \{A\})$

$V = \{V1, V2, V3, V4\}$

$A = \{<V1, V2>, <V1, V3>, <V3, V4>, <V4, V1>\}$

$<w, v>$

入边：

$G1 = (V, \{A\})$

$V = \{V1, V2, V3, V4\}$

$A = \{<V4, V1>, <V2, V1>, <V3, V1>, <V4, V3>$

B> 表示无向图

$G = (V, \{E\})$

$V = \{v1, v2, v3, v4, v5\}$

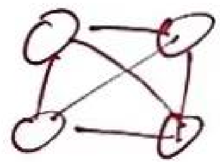
$E = \{(v1, v2), (v1, v4), (v2, v3), (v2, v5), (v3, v4), (v3, v5)$

$\}$

顶点和边（弧）的关系.

前提 <
- n: 顶点个数.
- e: 边（弧）的数目.

① 无向图: e取值: $\boxed{0 \sim \frac{n}{2}(n-1)}$



☆ 完全图: 有 $\frac{1}{2}n(n-1)$ 条边的无向图.

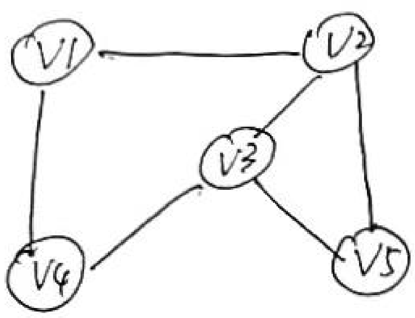
⟶ 任何两个顶点之间都有一条边

② 有向图: e取值: $\boxed{0 \sim n(n-1)}$

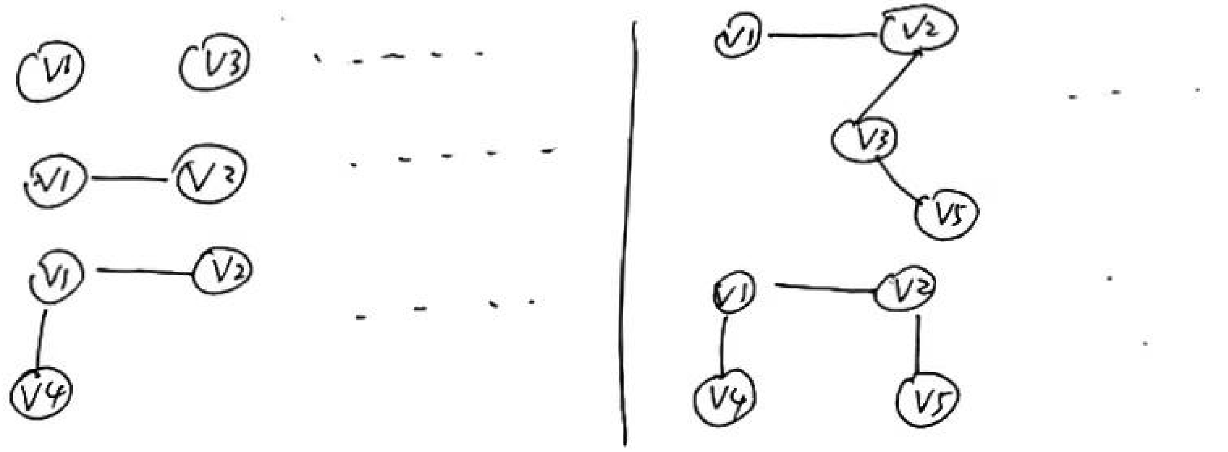
有向完全图: 有 $n(n-1)$ 条边的有向图.

⟶ 任何两个顶点之间都有一条出边
一条入边.

③ 稀疏图.



部分：

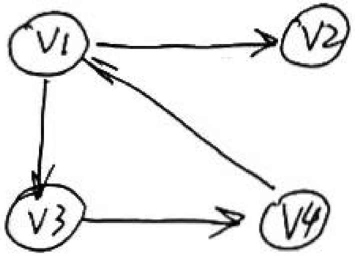


④ 子图.

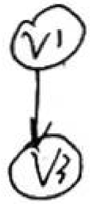
设有两个图 G. G'. $G=(V,\{E\})$. $G'=(V',\{E'\})$.

如果. $V'\subseteq V$, $E'\subseteq E$. 则称 G'为 G 的子图.

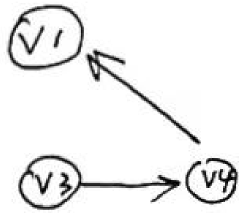
例.



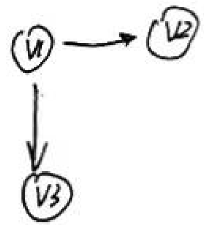
子图.

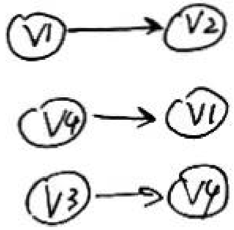


对.



对





⑤ 邻接点.

A> 无向图:

(顶点V、顶点W 互为邻接点)

☆ 无向图顶点的度:

TD(V)= 和顶点V相关联的边的数目.

⇒ 无向图顶点的度之和是边的2倍.

B> 有向图邻接点.

W邻接自V

V邻接到W

(W是V的出边邻接点) √

(V是W的入边邻接点) √

有向图顶点的度:

i> 入度 ID(V): 以顶点V为头的弧的数目. (入边的数目).

ii> 出度 OD(V): 以顶点V为尾的弧的数目. (出边的数目)

TD(V) = ID(V) + OD(V).

☆ 有n个顶点、e条弧(边)的图满足:

$$\boxed{e = \frac{1}{2}\sum_{i=1}^{n} TD(V_i)}$$

(边的个数即所有顶点的度之和的一半)

① 路径:

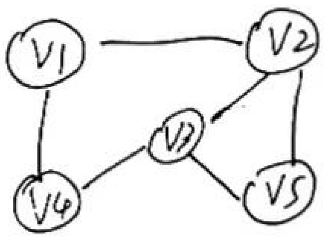
路径长度: 从 V_i 到 V_j 的一条路上边的数目.

回路(环): 从 V_i 到 V_j 的一条路径上出现了重复顶点.

简单路径: 路径中的顶点不重复.

☆ 简单回路(环):

除首尾顶点外, 其余顶点不重复.



i> 从 V_1 到 V_5 路径:

$V_1 \to V_2 \to V_5$

$V_1 \to V_2 \to V_3 \to V_5$

$V_1 \to V_4 \to V_3 \to V_5$

$V_1 \to V_4 \to V_3 \to V_2 \to V_5$

简单路径

ii>

$V_1 \to V_2 \to V_3 \to V_4 \to V_1 \to V_2 \to V_5$

—— 回路

简单回路:

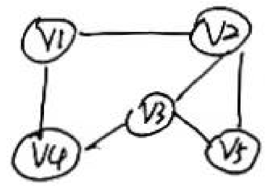
① $V_1 \to V_2 \to V_3 \to V_4 \to V_1$

② $V_2 \to V_3 \to V_5 \to V_2$

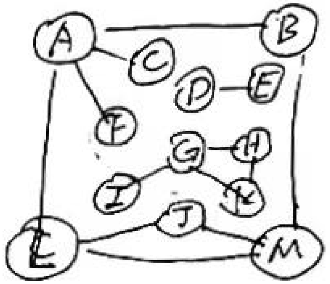
☆ ⑧ 无向图之连通图.

A> 连通：无向图中任意两个顶点都有路径，称这两个顶点是连通的

B> 连通图：无向图任意两个顶点都是连通的

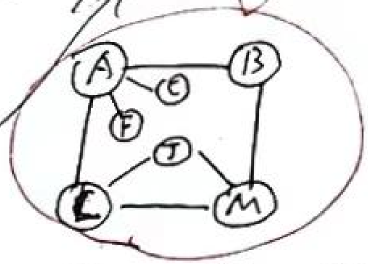


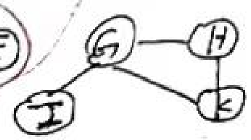
（连通图）

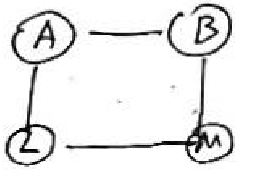


（不是连通图）

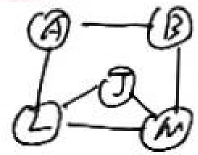
连通分量 —— 无向图的极大连通子图







（×）



（×）

不是连通分量. 而 该无向图可能含有多个连通分量.

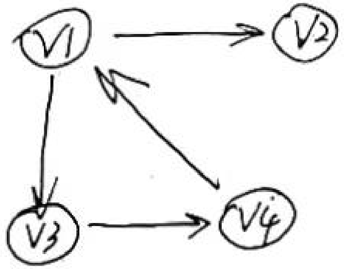
i> 若无向图本身是连通图 那无向图即为它的连通分量.

ii> 若无向图非连通图.

↳ 满足 | 是原图的一个子图.
| 满足该子图是连通的
| 极大连通子图 —— 不可以再加一个顶点. 构成连通图.

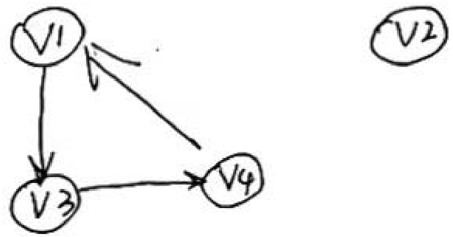
⑨ 有向图：强连通图

—— 在有向图中，每一对 $V_i, V_j \in V$，$V_i \neq V_j$
从 V_i 到 V_j 和从 V_j 到 V_i 都存在路径。



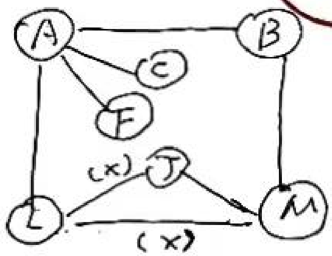
不是强连通图（例：V_2 到 V_1 无路）

强连通分量：有向图中的极大强连通子图。

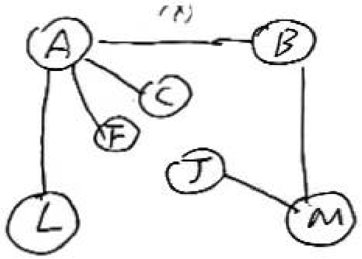


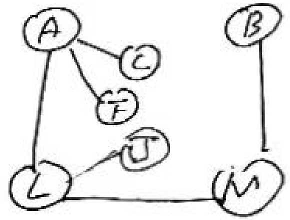
10> 连通图的生成树 —— 一个连通图的极小连通子图。

(答案不唯一)



⇒





(连通分量)

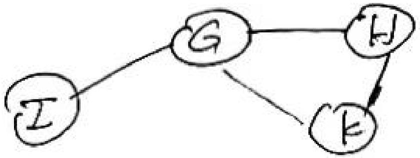
⇒ 极大连通子图

☆ 只有连通图才有生成树

☆ 如果在一棵生成树上加上任何一条边，都必定构成一个环。

(连通图所有顶点)

☆ 一棵生成树要包含所有顶点，但只有构成它的 n-1 条边。



⇒

性质：i> 一棵有 n 个顶点的生成树有且仅有 n-1 条边。

ii> 一个图有 n 个顶点和小于 n-1 条边，(有顶点无法连通)

—— 则该图一定是一个非连通图

iii> 若多于 n-1 条边 —— 该图带环

☆ 有 n-1 条边的~~生成树~~图不一定是生成树。