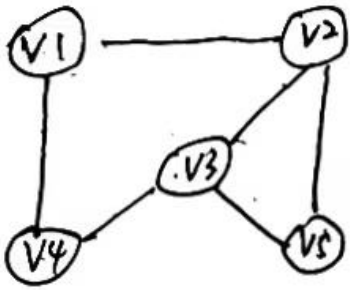
图的存储结构

① 邻接矩阵存储 —— 二维数组存储

以顶点构建邻接矩阵(行、列)

A> 无向图



	V1	V2	V3	V4	V5
V1	0	1	0	1	0
V2	1	0	1	0	1
V3	0	1	0	1	1
V4	1	0	1	0	0
V5	0	1	1	0	0

0 —— 无边.

1 —— 有边.

对角线上的值为0

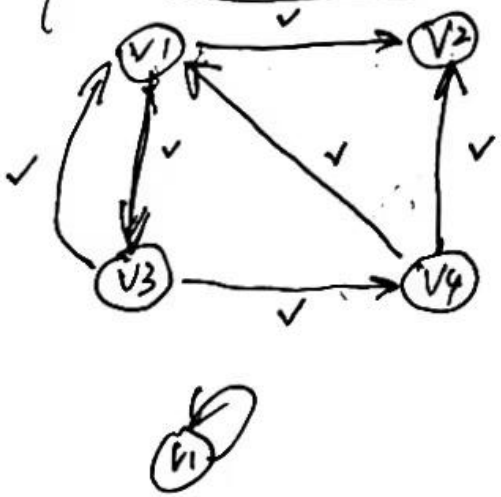
表示自己和自己无边.

矩阵中"1"的个数是边的二倍

行 —— 出边.

时间复杂度: 列: $O(n^2)$ $O(n^2)$

B> 有向图

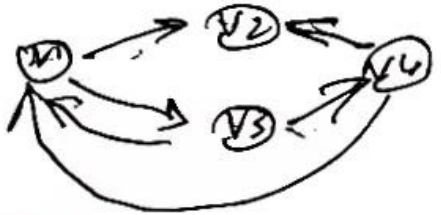


	V1	V2	V3	V4
V1	0	1	1	0
V2	0	0	0	0
V3	1	0	0	1
V4	1	1	0	0

行 表示出边

列 表示入边

"1"的个数即弧的值



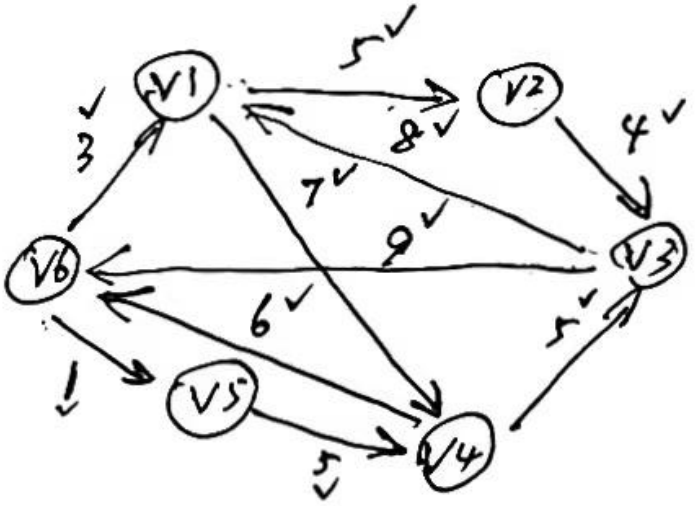
时: $O(n^2)$

③ 网 —— 是图的一种应用

—— 边(弧)上有权值的图.

1> 存储:有向图有向图.

2> 值 | 0 —— 结点自己去也
和值 — 有边.
∞ — 两个顶点无边.



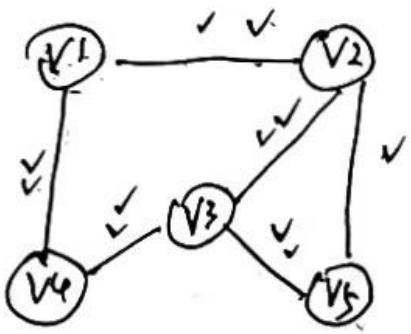
	V1	V2	V3	V4	V5	V6
V1	0	5	∞	7	∞	∞
V2	∞	0	4	∞	∞	∞
V3	8	∞	0	∞	∞	9
V4	∞	∞	5	0	∞	6
V5	∞	∞	∞	5	0	∞
V6	3	∞	∞	∞	1	0

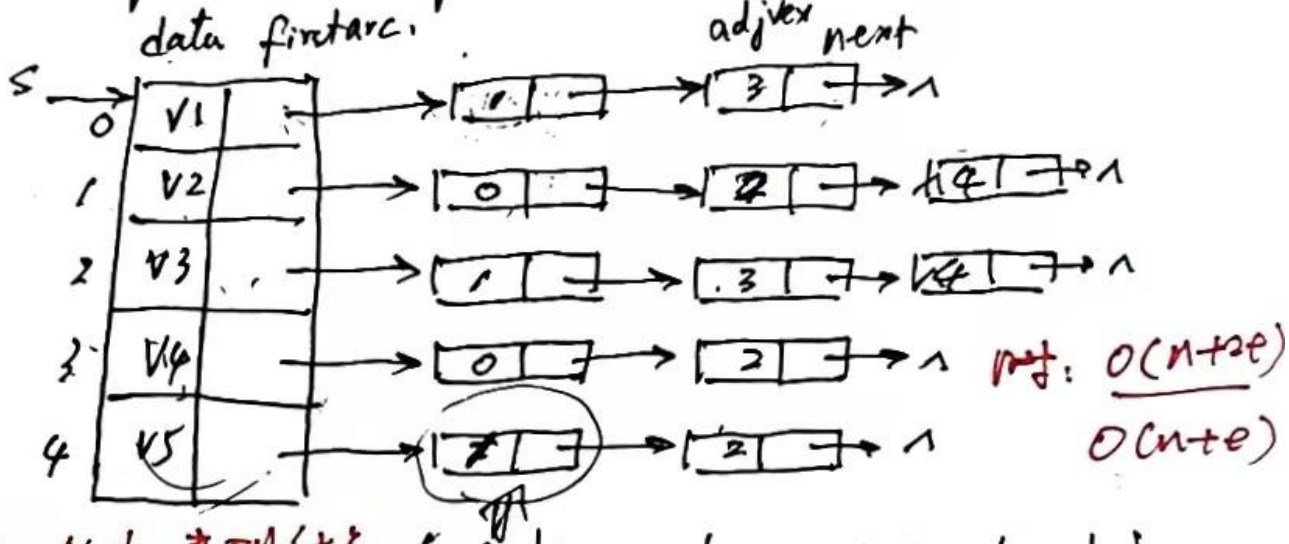
合个邻矩阵存储.

② 时间复杂度: $O(n^2)$

邻接表存储

① 无向图：





☆ ArcNode 表现结点 ArcNode 与结点也的之后

结构定义

邻接点表现：

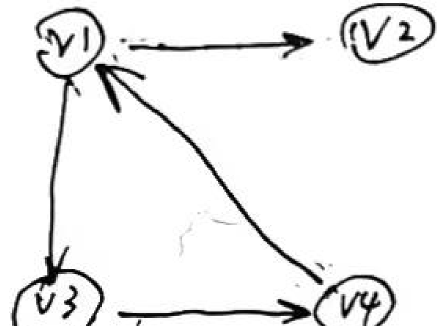
```
typedef struct ArcNode{
    int adjvex; // 存放邻接点下标
    struct ArcNode *next; // 指向下一个邻接点
}ArcNode;
```

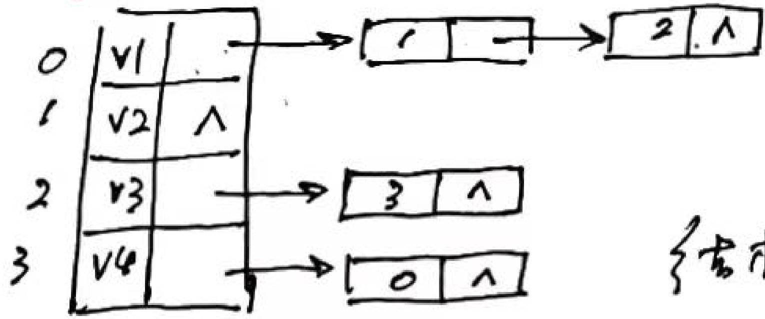
```
typedef struct{
    Type data; // 存储顶点信息的表现
    ArcNode *firstarc;
    // 存放：指向第一个邻接点表结点的头指针
}
```

② 有向图：< 出边邻接表 / 入边邻接表

结点个数即弧的个数

出边邻接表.



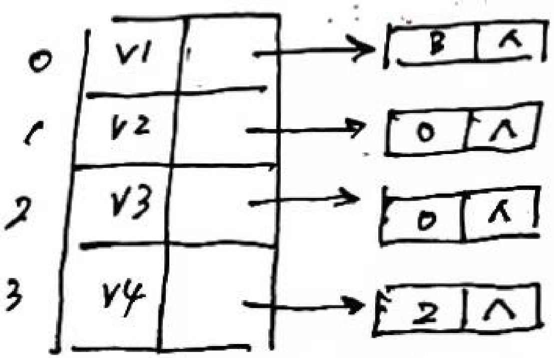


O(n+e)

结构定义同无向图

邻接表 数据保存下 ⇓ 无法得到

入边邻接表



缺陷：

在出边邻接表中无法得到

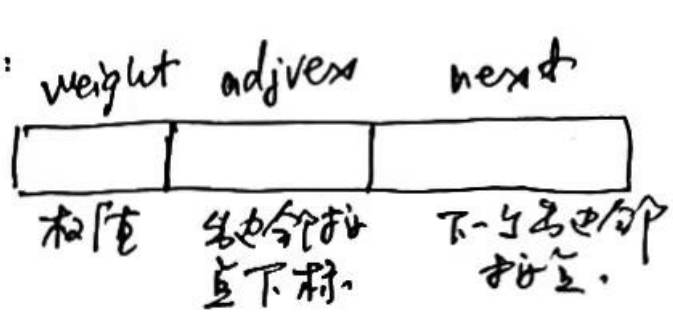
一个结点的入边邻接点信息

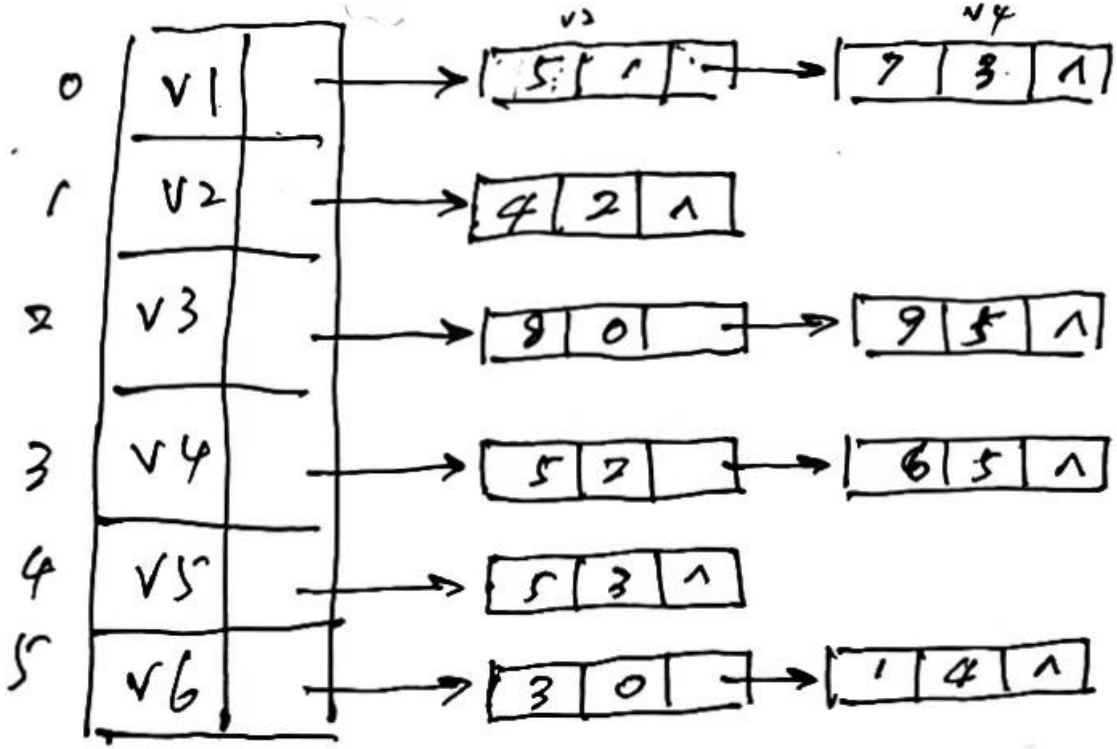
☆ 网的邻接表存储：在结点类型中加一个权值(weight)域

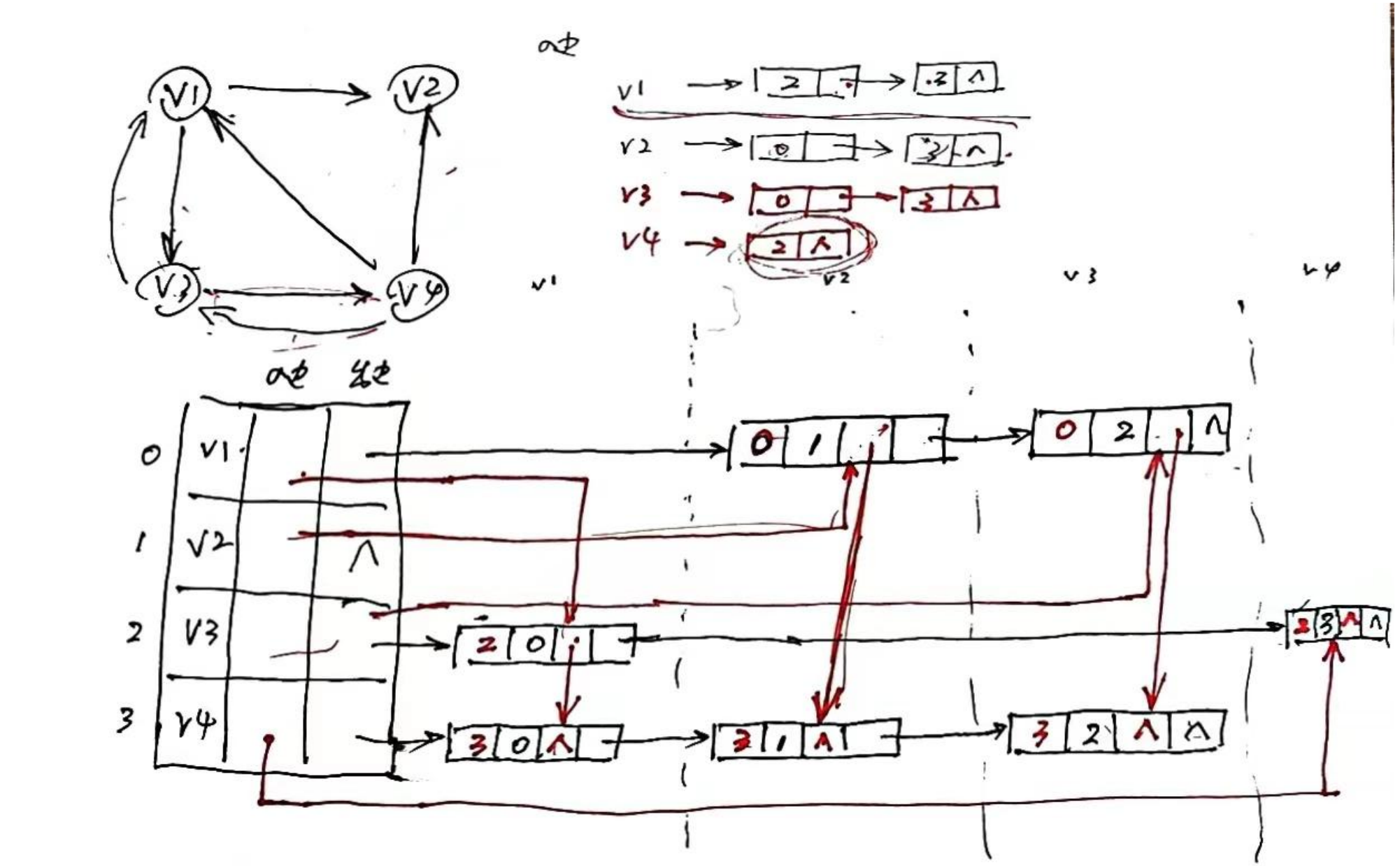
☆ 邻接表存储遍历邻接点的时间复杂度 O(n+e)

网的邻接表：

结点：


weight
adjvex
next
权值
邻接点下标
下一个邻接点指针


0
1
2
3
4
5
V1
V2
V3
V4
V5
V6
V3
V4
5 1
7 3 ∧
4 2 ∧
3 0
9 5 ∧
5 7
6 5 ∧
5 3 ∧
3 0
1 4 ∧


V1
V2
V3
V4
0
1
2
3
∧

图的遍历

1. 定义：从某个顶点出发遍历图中的所有顶点，
且每个顶点只被访问一次
（找到一条搜索路径，将图中所有顶点进行线性排列）

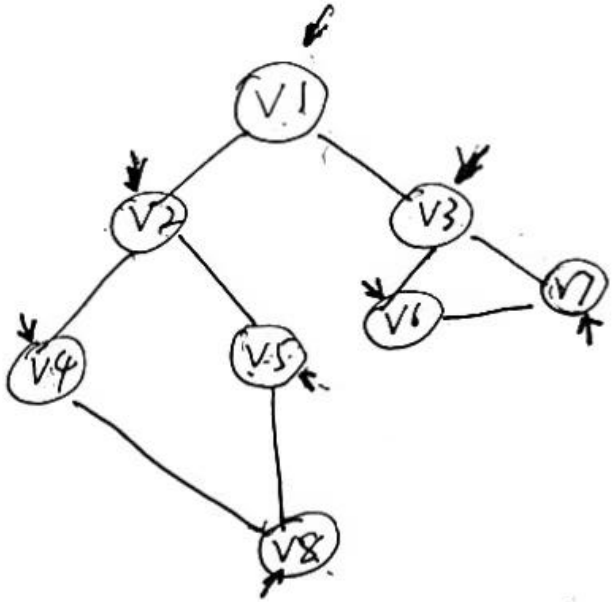
作用：
i> 判断图的连通性问题
ii> 拓扑排序
iii> 关键路径

☆ 图的遍历因出发点的起始顶点不同，
线性排列不同，
——遍历结果不一定相同。

方式
- 深度优先遍历 (DFS)
- 广度优先遍历 (BFS)

深度优先遍历 —— 类似于二叉树的前序遍历

以从顶点 v1 开始为例：



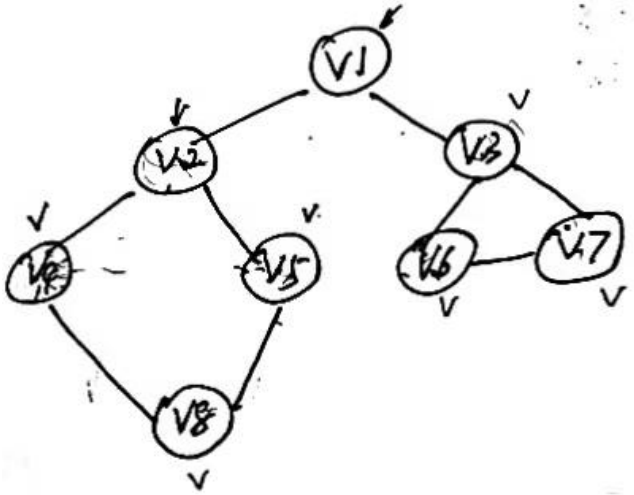
v1 → v2 → v4 → v8 → v5 → v3 → v6 → v7

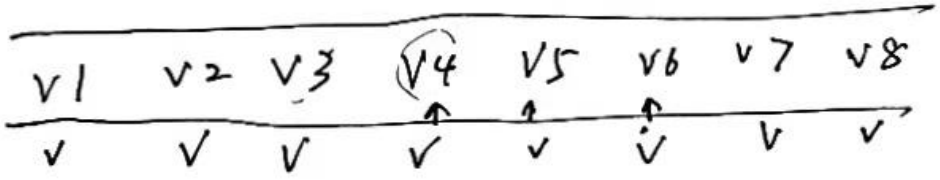
结论：是一个递归过程。

二叉树的按层——

广度优先遍历 —— 类似于队列方式

v1 → v2 → v3 → v4 → v5 → v6 → v7 → v8

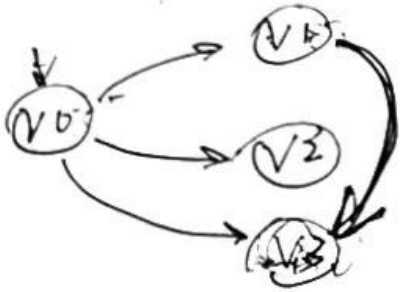




2015-1-⑤ $V = \{v0, v1, v2, v3\}$

$E = \{\langle v0, v1\rangle \langle v0, v2\rangle \langle v0, v3\rangle \langle v1, v3\rangle\}$

图的广度遍历：

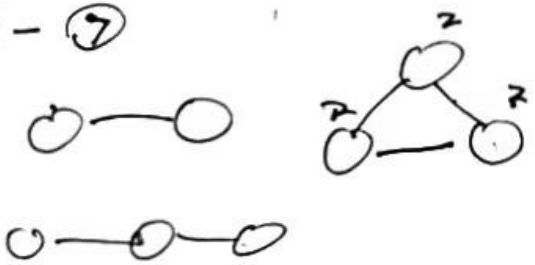


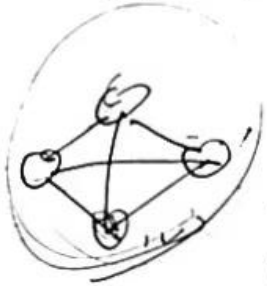
v0 → v1 → v3 → v2

v0 → v2 → { v1 → v3 ; v3 → v1 }

v0 → v3 → { v1 → v2 ; v2 → v1 }

2009-1-⑦

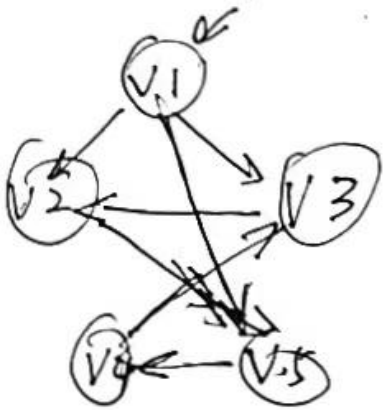




$\frac{1}{2}n(n-1)$

$1 - \frac{1}{2}n(n-1)$

2016-1-6 课时



V1 V5 V4 V3 V2 ✓

V1 V3 V2 V5 V4 ✓

V1 V2 V5 V4 V3 ✓

V1 V2 V3 V4 V5 ×

V1 → V5 → V4 → V3 → V2

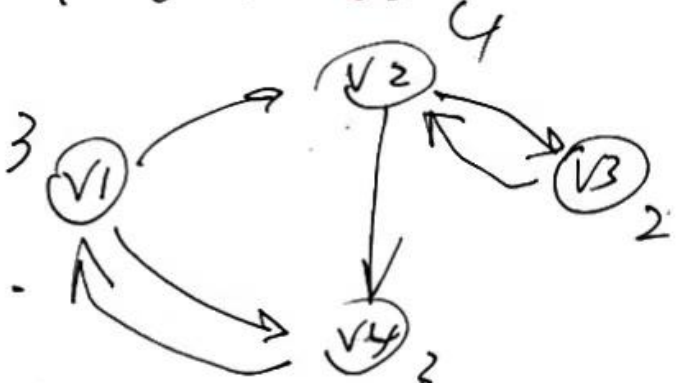
2007-1-2

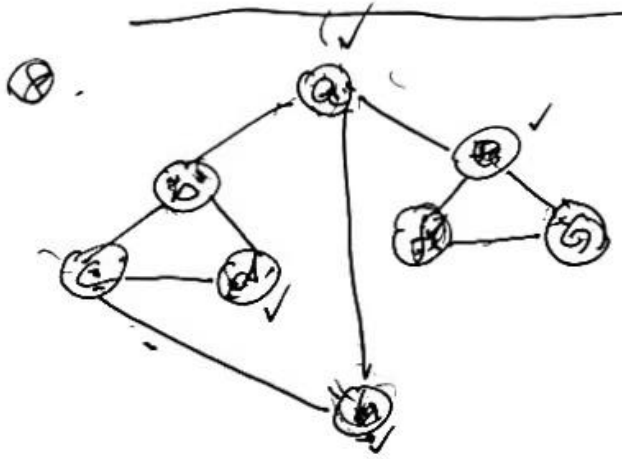
$$A=\begin{bmatrix} 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

因为不对称

—— 有向、无向图

—— 非有向图





(√)

h √ c √ a √ b √ d √ e √ g √ f √

(√)

e √ a √ f √ g √ b √ h √ c √ d √

(√)

d √ b √ c √ a √ h √ e √ f √ g √

(×).

a b c d h e f g