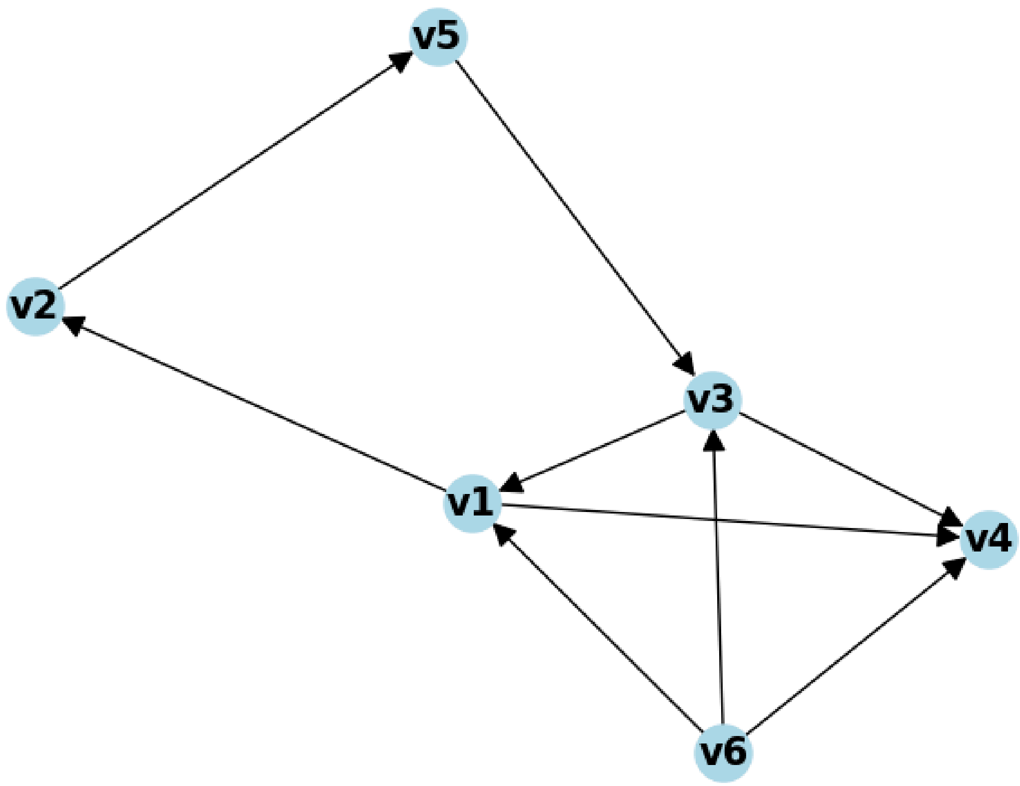
离散（2）hw2

> 王子轩 `2023011307`
>
> `wang-zx23@mails.tsinghua.edu.cn`

P14 T16

> 写出如图的邻接矩阵、关联矩阵、边列表、正向表




```
import networkx as nx
import matplotlib.pyplot as plt
DG = nx.DiGraph()
DG.add_node("v1")
DG.add_node("v2")
DG.add_node("v3")
DG.add_node("v4")
DG.add_node("v5")
DG.add_node("v6")
DG.add_edge("v1", "v2")
DG.add_edge("v1", "v4")
DG.add_edge("v2", "v5")
DG.add_edge("v3", "v1")
DG.add_edge("v3", "v4")
DG.add_edge("v5", "v3")
DG.add_edge("v6", "v1")
DG.add_edge("v6", "v3")
DG.add_edge("v6", "v4")
```

```python
nx.draw(DG, with_labels=True, font_weight='bold', node_size=500,
node_color='lightblue', font_size=15, arrowsize=20)
```

```
{'v1': ['v2', 'v4'],
'v2': ['v5'],
'v3': ['v1', 'v4'],
'v4': [],
'v5': ['v3'],
'v6': ['v1', 'v3', 'v4']}
```

- 邻接矩阵

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

- 关联矩阵

$$
\mathbf{B}=\begin{bmatrix}1&1&0&-1&0&0&-1&0&0\\-1&0&1&0&0&0&0&0&0\\0&0&0&1&1&-1&0&-1&0\\0&-1&0&0&-1&0&0&0&-1\\0&0&-1&0&0&1&0&0&0\\0&0&0&0&0&0&1&1&1\end{bmatrix}
$$

- 边列表

$$
\begin{aligned}\mathbf{A}&=\begin{bmatrix}1&1&2&3&3&5&6&6&6\end{bmatrix}\\\mathbf{B}&=\begin{bmatrix}2&4&5&1&4&3&1&3&4\end{bmatrix}\end{aligned}
$$

- 正向表

$$
\begin{aligned}\mathbf{A}&=\begin{bmatrix}1&3&4&6&6&7&10\end{bmatrix}\\\mathbf{B}&=\begin{bmatrix}2&4&5&1&4&3&1&3&4\end{bmatrix}\end{aligned}
$$

P14 T17

> 判断两图是否同构

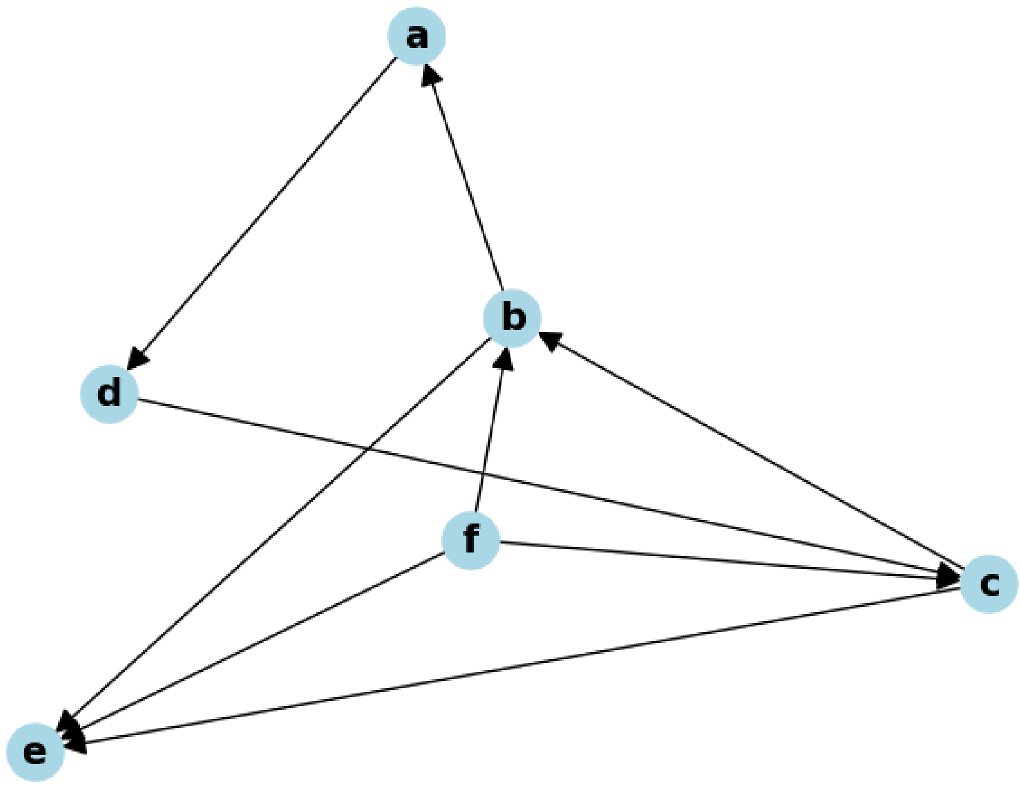


图1 $G_1=(V_1,E_1)$和$G_2=(V_2,E_2)$同构，存在$V_1\leftrightarrow V_2:f$，$(v_i,v_j)\in G_1\Leftrightarrow(f(v_i),f(v_j)\in G_2$

$$
\begin{aligned}
f(v1)&=b\\
f(v2)&=a\\
f(v3)&=c\\
f(v4)&=e\\
f(v5)&=d\\
f(v6)&=f
\end{aligned}
$$

P14 T21

> 表示一个n个顶点, m条边的非赋权图需要多少存储空间:分别对邻接矩阵\关联矩阵\边列表\正向表进行分析

- 邻接矩阵：需要存储n^2的矩阵，空间复杂度为$\mathcal{O}(n^2)$
- 关联矩阵：需要存储$n\times m$的矩阵，空间复杂度为$\mathcal{O}(nm)$
- 边列表：每条边需要维护两个顶点的编号，共m条边，则需要存储大小为$2m$个顶点编号，空间复杂度为$\mathcal{O}(m)$
- 正向表：采用一个索引数组A长度为$n+1$记录每个顶点的邻接列表起始位置，一个后继数组B长度为m记录每个顶点的直接后继，空间复杂度为$\mathcal{O}(n+m)$