Algoritmos e Estruturas de Dados II
Aula 02 – Grafos: Conceitos Básicos

Prof. Luciano A. Digiampietri
digiampietri@usp.br
@digiampietri

Grafos – Conceitos Básicos[1]

- O que é um grafo?

[1] Slides baseados no material do prof. Norton Trevisan Roman (disciplina Estrutura de Dados - UNIVESP: https://www.youtube.com/playlist?list=PLxI8Can9yAHf8k8LrUePyj0y3lLpigGcl)

Grafos – Conceitos Básicos[1]

- O que é um grafo?
- Grafos são estruturas matemáticas (ou modelos matemáticos) que permitem codificar relacionamentos entre pares de objetos

[1] Slides baseados no material do prof. Norton Trevisan Roman (disciplina Estrutura de Dados - UNIVESP: `https://www.youtube.com/playlist?list=PLxI8Can9yAHf8k8LrUePyj0y3lLpigGcl`)

Grafos – Conceitos Básicos[1]

- O que é um grafo?
- Grafos são estruturas matemáticas (ou modelos matemáticos) que permitem codificar relacionamentos entre pares de objetos
  - Os objetos são os vértices do grafo

[1] Slides baseados no material do prof. Norton Trevisan Roman (disciplina Estrutura de Dados - UNIVESP: https://www.youtube.com/playlist?list=PLxI8Can9yAHf8k8LrUePyj0y3lLpigGcl)

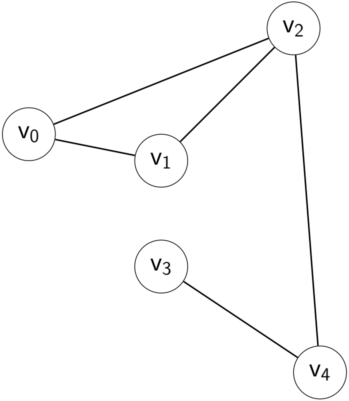
Grafos – Conceitos Básicos[1]

- O que é um grafo?
- Grafos são estruturas matemáticas (ou modelos matemáticos) que permitem codificar relacionamentos entre pares de objetos
  - Os objetos são os vértices do grafo
  - Os relacionamentos são suas arestas

[1]Slides baseados no material do prof. Norton Trevisan Roman (disciplina Estrutura de Dados - UNIVESP: https://www.youtube.com/playlist?list=PLxI8Can9yAHf8k8LrUePyj0y31LpigGcl)

Grafos – Conceitos Básicos

- São representados como um conjunto de nós (vértices) conectados par a par por linhas (arestas)

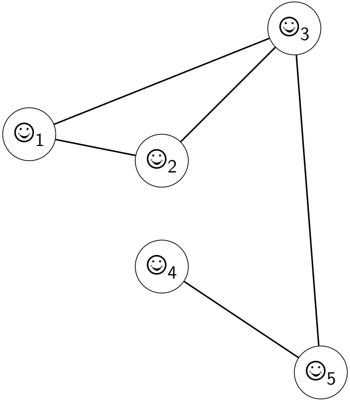


Grafos – Conceitos Básicos

- Podem ser utilizados para representar uma infinidade de situações/problemas

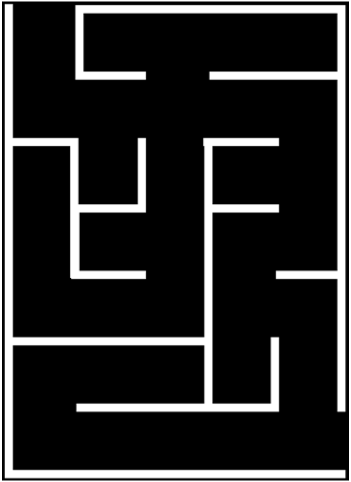
Grafos – Conceitos Básicos

- Podem ser utilizados para representar uma infinidade de situações/problemas
- Podem modelar conexões em <u>redes sociais</u>



Grafos – Conceitos Básicos

- Podem ser utilizados para representar uma infinidade de situações/problemas
- Podem modelar conexões em redes sociais
- Labirintos

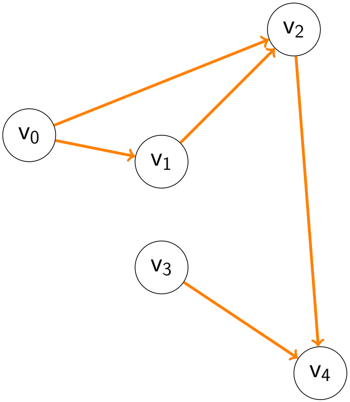


Grafos – Conceitos Básicos

- Podem ser utilizados para representar uma infinidade de situações/problemas
- Podem modelar conexões em <u>redes sociais</u>
- Labirintos
- Rotas de metrô

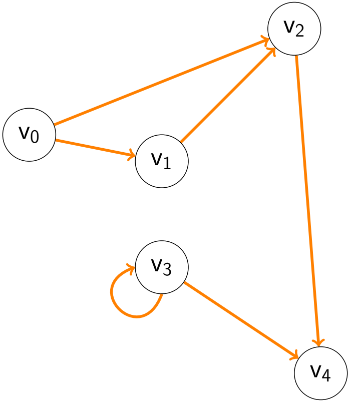
Grafos – Conceitos Básicos

- Alguns grafos são dirigidos (ou direcionados) - digrafos
  - As relações representadas pelas arestas têm sentido definido
  - As arestas só podem ser seguidas em uma única direção



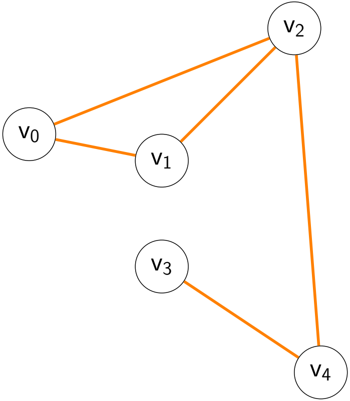
Grafos – Conceitos Básicos

- Em grafos dirigidos, as arestas são pares ordenados de vértices
  - Saindo de um em direção ao outro
  - Mesmo que ambos sejam o mesmo vértice (auto-laços - *self-loop*)



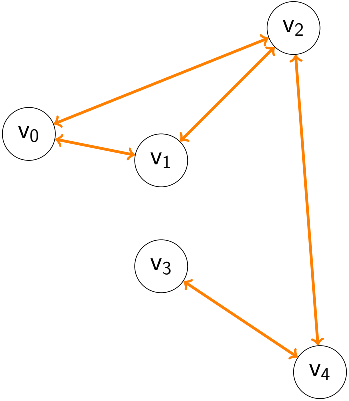
Grafos – Conceitos Básicos

- Outros são <u>não dirigidos</u> (ou <u>não direcionados</u>)
  - As relações representadas pelas arestas não têm sentido definido
  - As arestas podem ser seguidas em qualquer direção



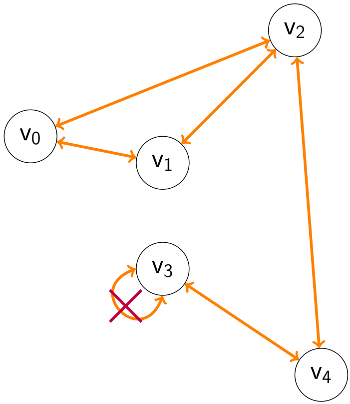
Grafos – Conceitos Básicos

- Podemos pensar num grafo não dirigido como um grafo dirigido com arestas de sentido duplo
- As arestas são pares não ordenados de vértices
- *Self-loops* não são permitidos



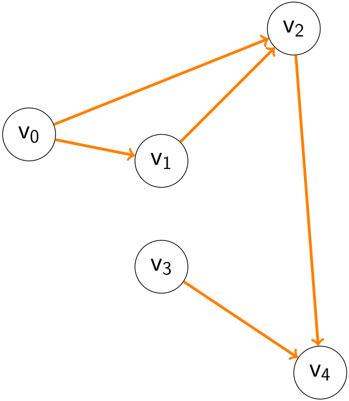
Grafos – Conceitos Básicos

- Podemos pensar num grafo não dirigido como um grafo dirigido com arestas de sentido duplo
- As arestas são pares não ordenados de vértices
- *Self-loops* não são permitidos



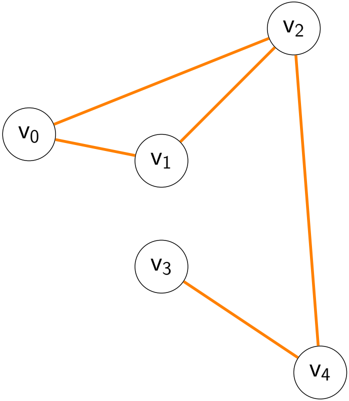
Grafos – Conceitos Básicos

- Se (u, v) é uma aresta no grafo, então dizemos que v é adjacente a u
  - Alternativamente, que v é vizinho de u
- (u, v) significa que a aresta sai de u e entra em v



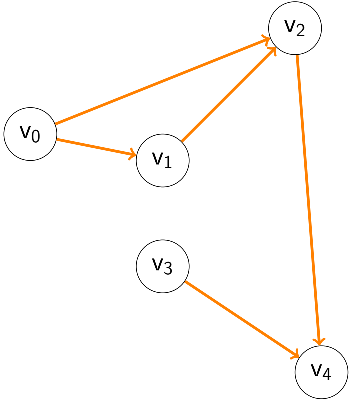
Grafos – Conceitos Básicos

- Em grafos não dirigidos, a relação de adjacência é simétrica
  - $(u, v) \Leftrightarrow (v, u)$



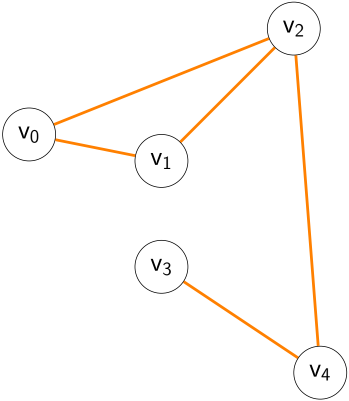
Grafos – Conceitos Básicos

- Em grafos não dirigidos, a relação de adjacência é simétrica
  - $(u, v) \Leftrightarrow (v, u)$
- Já em dirigidos, não necessariamente há tal simetria
  - Há (v_0, v_1), mas não (v_1, v_0)



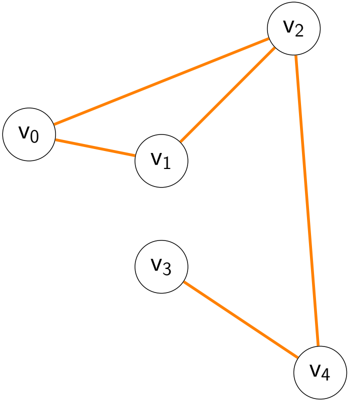
Grafos – Conceitos Básicos

- Em grafos não dirigidos, o grau de um vértice é o número de arestas que incidem nele



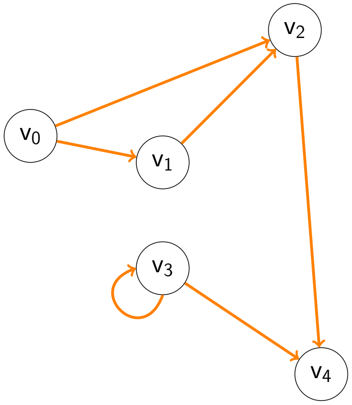
Grafos – Conceitos Básicos

- Em grafos não dirigidos, o grau de um vértice é o número de arestas que incidem nele
  - $gr(v_0) = gr(v_1) = gr(v_4) = 2$
  - $gr(v_2) = 3$
  - $gr(v_3) = 1$



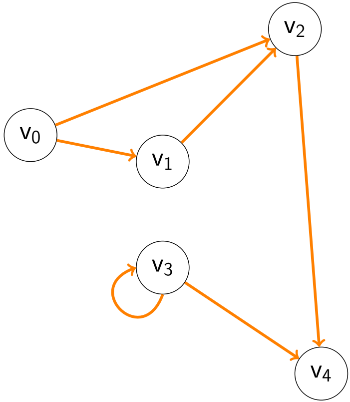
Grafos – Conceitos Básicos

- Já em grafos dirigidos, o grau de um vértice é o número de arestas que saem do vértice mais o número de arestas que chegam nele



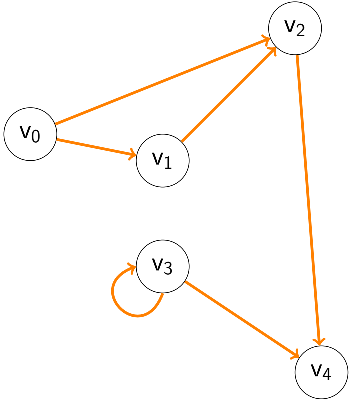
Grafos – Conceitos Básicos

- Já em grafos dirigidos, o grau de um vértice é o número de arestas que <u>saem</u> do vértice mais o número de arestas que <u>chegam</u> nele
  - $gr(v_0) = gr(v_1) = gr(v_4) = 2$
  - $gr(v_2) = gr(v_3) = 3$



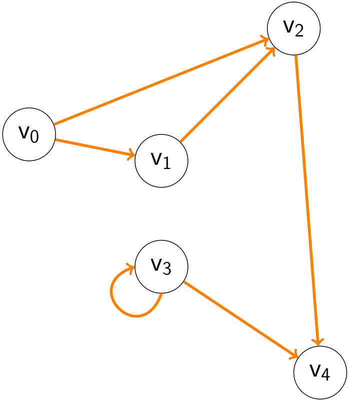
Grafos – Conceitos Básicos

- No caso de grafos dirigidos, há dois tipos específicos de graus de vértice:



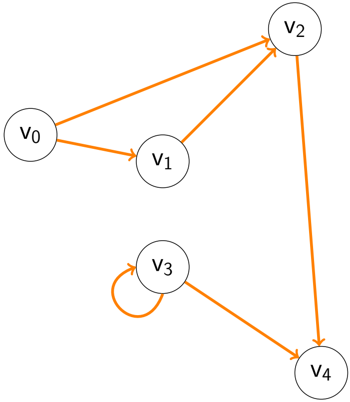
Grafos – Conceitos Básicos

- No caso de grafos dirigidos, há dois tipos específicos de graus de vértice:
  - Grau de saída: número de arestas que saem do vértice
  - Grau de entrada: número de arestas que chegam no vértice



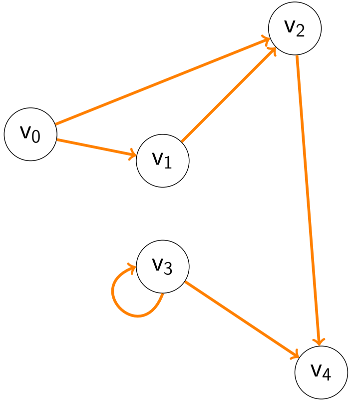
Grafos – Conceitos Básicos

- Um caminho de um vértice x a um vértice y é uma sequência de vértices em que, para cada vértice, do primeiro ao penúltimo, há uma aresta ligando esse vértice ao próximo na sequência.



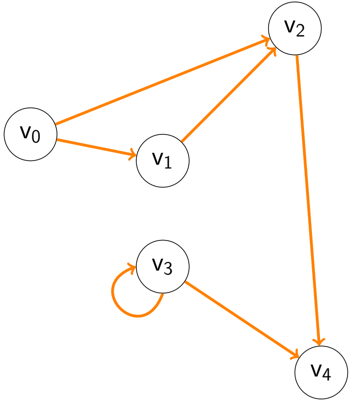
Grafos – Conceitos Básicos

- No caso ao lado, alguns caminhos são:



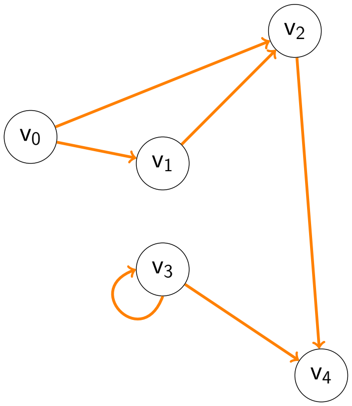
Grafos – Conceitos Básicos

- No caso ao lado, alguns caminhos são:
  - (v_0, v_1, v_2, v_4)
  - (v_3, v_4)
  - (v_0, v_1, v_2)
  - (v_3, v_3, v_4)



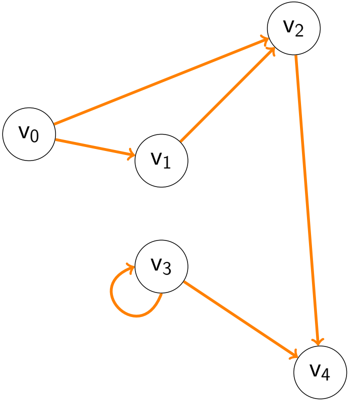
Grafos – Conceitos Básicos

- O comprimento de um caminho é o número de arestas nele



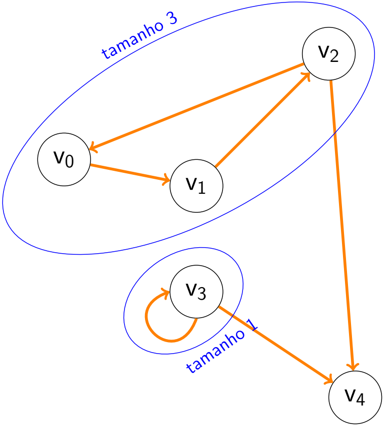
Grafos – Conceitos Básicos

- O comprimento de um caminho é o número de arestas nele
  - $compr(v_0, v_1, v_2, v_4) = 3$
  - $compr(v_3, v_4) = 1$
  - $compr(v_0, v_1, v_2) = 2$
  - $compr(v_3, v_3, v_4) = 2$



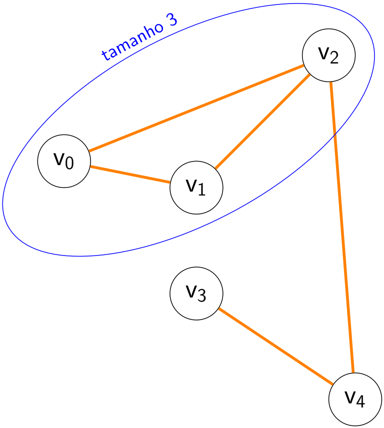
Grafos – Conceitos Básicos

- Um ciclo acontece quando, a partir de um determinado vértice, pudermos percorrer algum caminho que nos leve a esse mesmo vértice
  - Em grafos dirigidos, o caminho deve conter pelo menos uma aresta



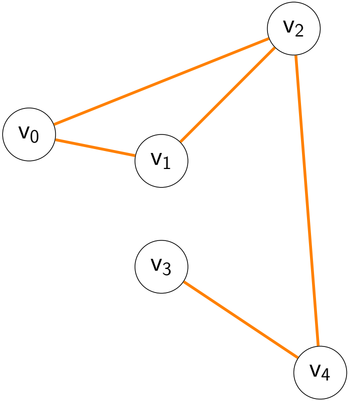
Grafos – Conceitos Básicos

- Em grafos não dirigidos, um ciclo deve conter pelo menos 3 arestas
- Grafos em que há ao menos um ciclo são chamados de <u>cíclicos</u>
- Grafos em que não há ciclos são chamados de <u>acíclicos</u>



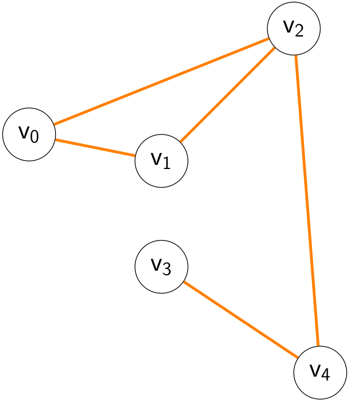
Grafos – Conceitos Básicos

- Um grafo não direcionado é conexo (ou conectado) se cada par de vértices nele estiver conectado por um caminho



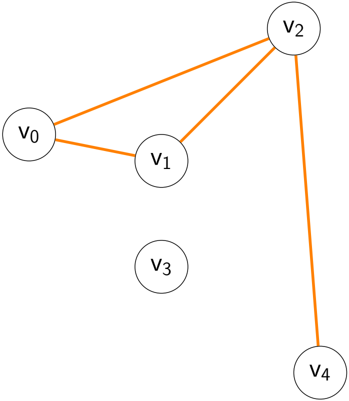
Grafos – Conceitos Básicos

- Um grafo não direcionado é conexo (ou conectado) se cada par de vértices nele estiver conectado por um caminho
  - O grafo ao lado é conexo



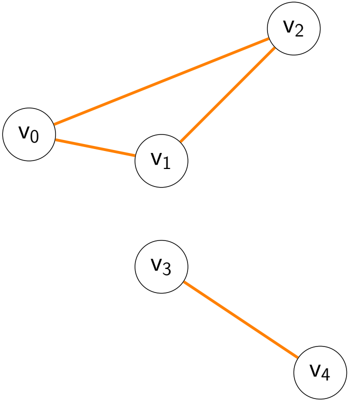
Grafos – Conceitos Básicos

- Um grafo não direcionado é conexo (ou conectado) se cada par de vértices nele estiver conectado por um caminho
  - O grafo ao lado é conexo
  - Agora é desconexo



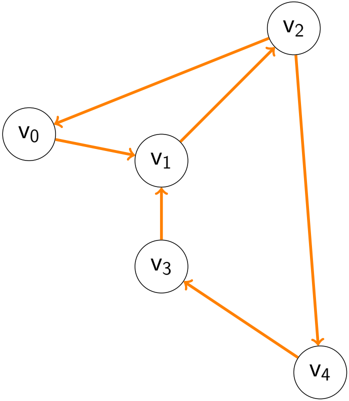
Grafos – Conceitos Básicos

- Um grafo não direcionado é <u>conexo</u> (ou <u>conectado</u>) se cada par de vértices nele estiver conectado por um caminho
  - O grafo ao lado é conexo
  - Agora é <u>desconexo</u>
  - E continua assim



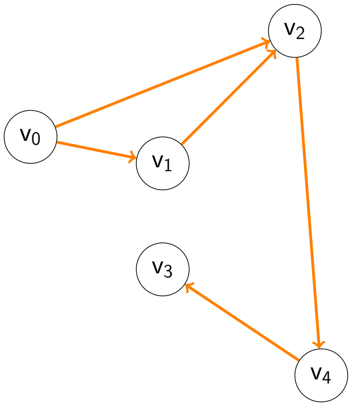
Grafos – Conceitos Básicos

- Um grafo dirigido é fortemente conexo se existir um caminho entre qualquer par de vértices no grafo
  - Contém um caminho direto de u para v e um caminho direto v para u para cada par de vértices (u, v)



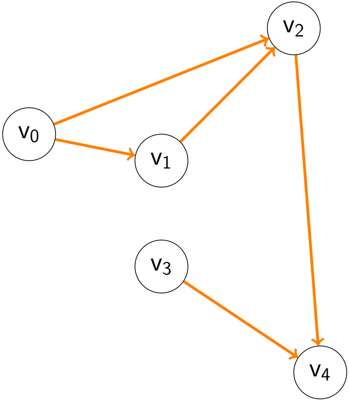
Grafos – Conceitos Básicos

- Um grafo dirigido é <u>conexo</u> se possuir um caminho de u para v, <u>ou</u> um caminho de v para u, para cada par de vértices (u, v)



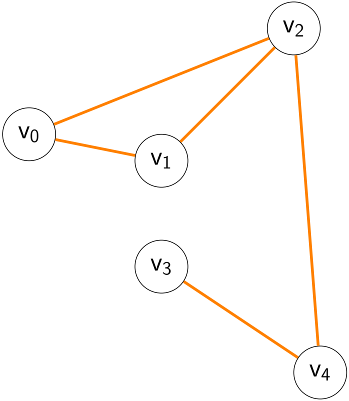
Grafos – Conceitos Básicos

- Um grafo dirigido é fracamente conexo se a substituição de todas as suas arestas por arestas não-direcionadas produz um grafo conexo.
  - Ex: não há caminho de $v_3 \rightarrow v_2$ nem de $v_2 \rightarrow v_3$ (não é conexo, mas é fracamente conexo)



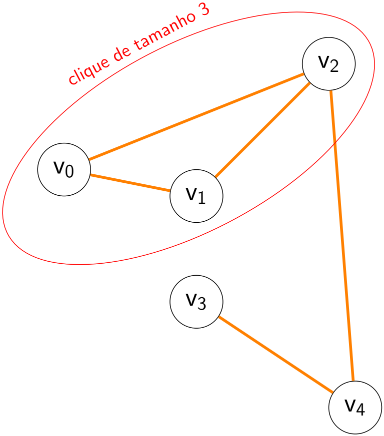
Grafos – Conceitos Básicos

- Em grafos não dirigidos, um clique é um subconjunto de seus vértices tal que cada par de vértices do subconjunto é conectado por uma aresta.



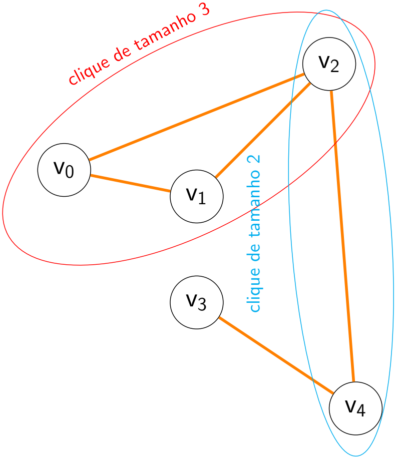
Grafos – Conceitos Básicos

- Em grafos não dirigidos, um clique é um subconjunto de seus vértices tal que cada par de vértices do subconjunto é conectado por uma aresta.



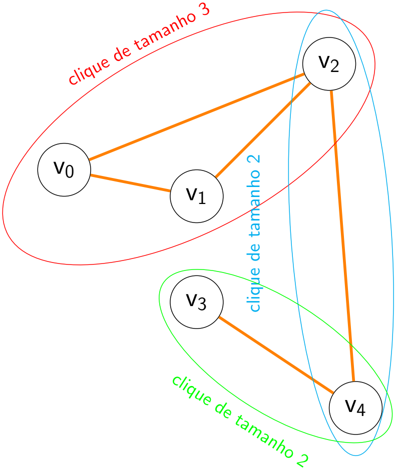
Grafos – Conceitos Básicos

- Em grafos não dirigidos, um clique é um subconjunto de seus vértices tal que cada par de vértices do subconjunto é conectado por uma aresta.



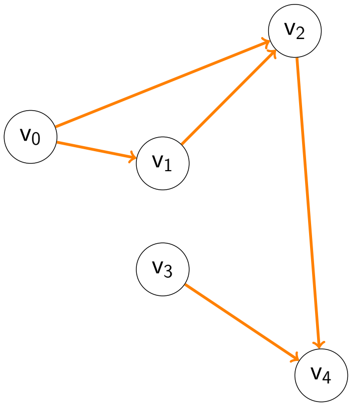
Grafos – Conceitos Básicos

- Em grafos não dirigidos, um clique é um subconjunto de seus vértices tal que cada par de vértices do subconjunto é conectado por uma aresta.



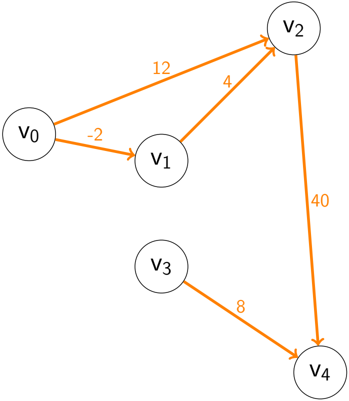
Grafos – Conceitos Básicos

- Grafos também podem ser ponderados



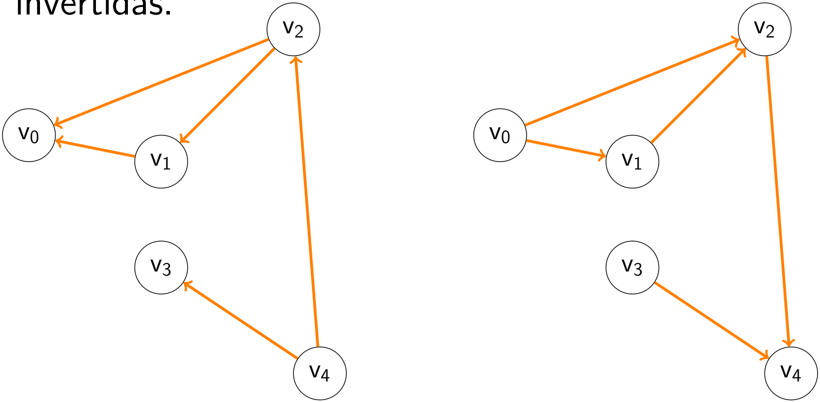
Grafos – Conceitos Básicos

- Grafos também podem ser ponderados
  - Caso em que possuem pesos associados às suas arestas
  - Esses pesos podem representar custos, distâncias etc.



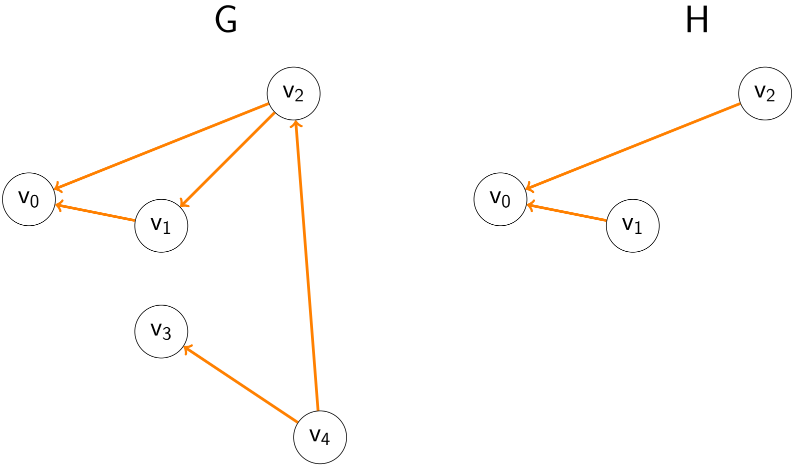
Grafos Transpostos

O grafo transposto G^T de G é um grafo que possui os mesmos vértices de G, mas as arestas têm suas direções invertidas.



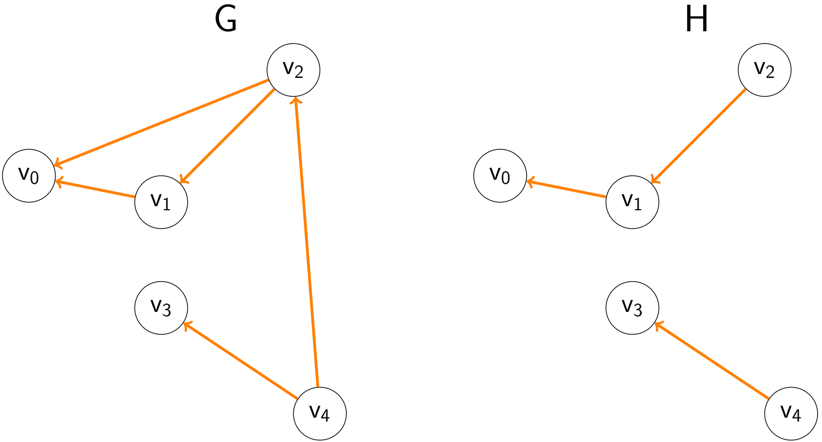
Subgrafos

Um grafo H é subgrafo de G se todo vértice e toda aresta de H também são vértices e arestas de G.



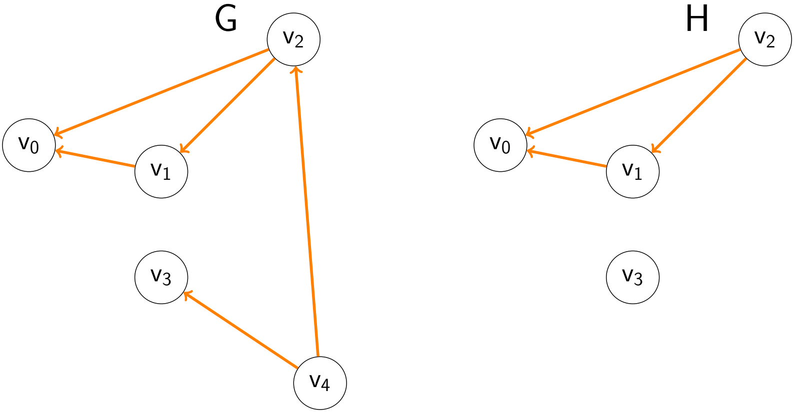
Subgrafo Gerador

Um subgrafo H de G é dito **gerador** (*spanning*) se contém todos os **vértices** de G.



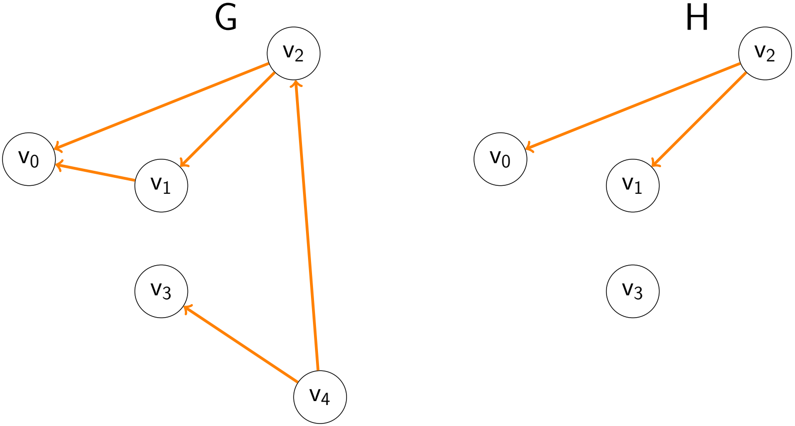
Subgrafo Induzido

Um subgrafo H é dito **induzido** de G se todas as **arestas** em G para pares de vértices que existem em H também existem em H.



Subgrafo Próprio

Um subgrafo H de G é dito **próprio** se ele é **diferente** de G.



Grafos Específicos

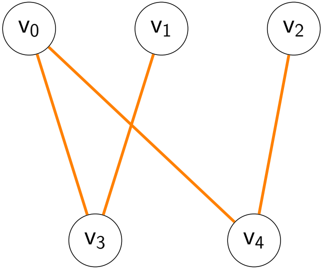
Existem, ainda, diversos tipos de grafos com especificidades em seus dados ou em sua representação.

Grafos Específicos

Existem, ainda, diversos tipos de grafos com especificidades em seus dados ou em sua representação.

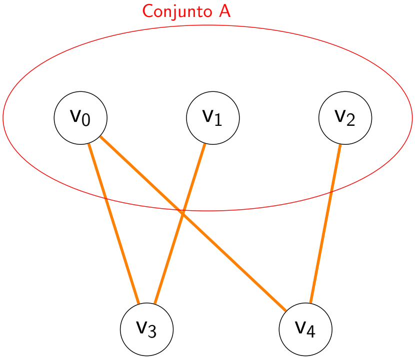
Grafo Bipartido

- Os vértices podem ser divididos em dois conjuntos disjuntos
- Não existem arestas entre vértices de um mesmo conjunto.



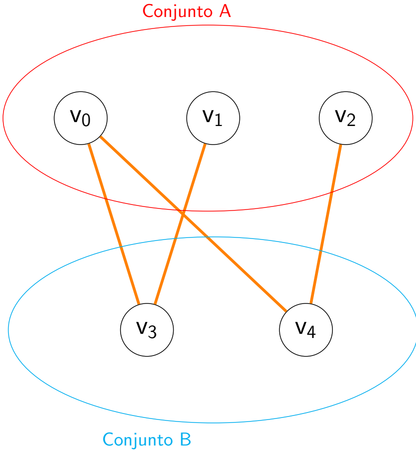
Grafo Bipartido

- Os vértices podem ser divididos em dois conjuntos disjuntos
- Não existem arestas entre vértices de um mesmo conjunto.



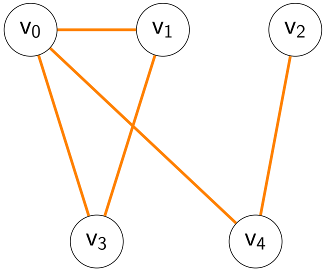
Grafo Bipartido

- Os vértices podem ser divididos em dois conjuntos disjuntos
- Não existem arestas entre vértices de um mesmo conjunto.



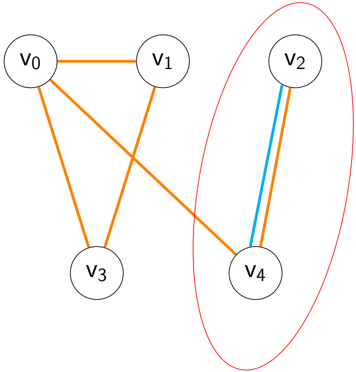
Multigrafo

- Permite a existência de arestas múltiplas entre um mesmo par de vértices.



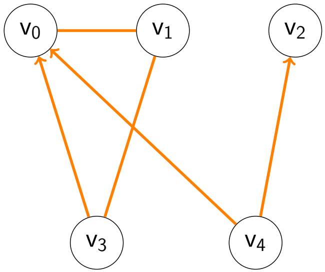
Multigrafo

- Permite a existência de arestas múltiplas entre um mesmo par de vértices.



Grafo Misto

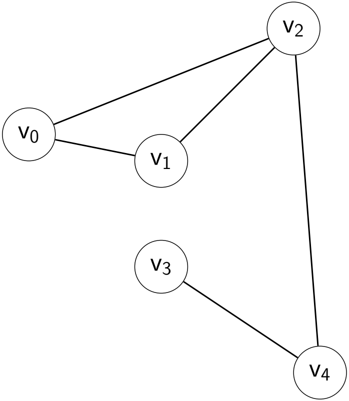
- Possui arestas direcionadas e não direcionadas.



Grafos – Representação

Voltando ao **Básico**:

- Grafos são representados como um conjunto de nós (vértices) conectados par a par por linhas (arestas)



Grafos – Representação

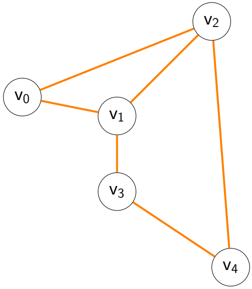
- Como podemos representar um grafo?

Grafos – Representação

- Como podemos representar um grafo?
  - Como um mapeamento de cada nó à lista de nós aos quais ele está conectado

Grafos – Representação

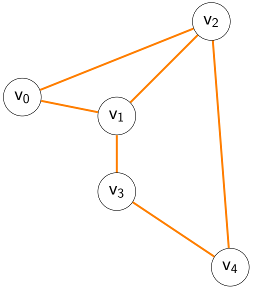
- Como podemos representar um grafo?
  - Como um mapeamento de cada nó à lista de nós aos quais ele está conectado



Não dirigido

Grafos – Representação

- Como podemos representar um grafo?
  - Como um mapeamento de cada nó à lista de nós aos quais ele está conectado

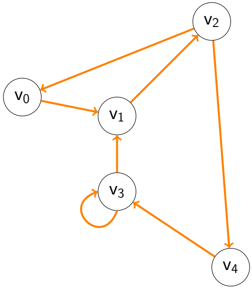


Não dirigido

Nó	*Conectado a*
v_0	$v_1,\ v_2$
v_1	$v_0,\ v_2,\ v_3$
v_2	$v_0,\ v_1,\ v_4$
v_3	$v_1,\ v_4$
v_4	$v_2,\ v_3$

Grafos – Representação

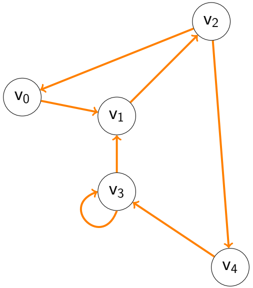
- Como podemos representar um grafo?
  - Como um mapeamento de cada nó à lista de nós aos quais ele está conectado



Dirigido

Grafos – Representação

- Como podemos representar um grafo?
  - Como um mapeamento de cada nó à lista de nós aos quais ele está conectado



Dirigido

Nó	*Conectado a*
v_0	v_1
v_1	v_2
v_2	v_0, v_4
v_3	v_1, v_3
v_4	v_3

Grafos – Representação

- Representação computacional de grafos

Grafos – Representação

- Representação computacional de grafos
- Existem duas maneiras usuais de representar grafos:
    - Matrizes de adjacência
    - Listas de adjacência

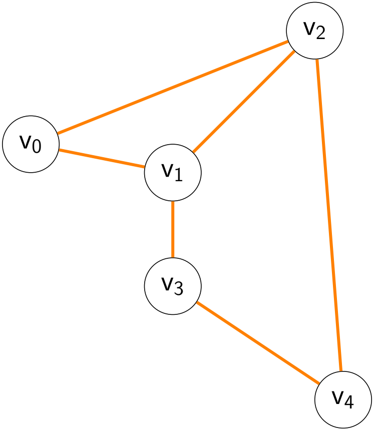
Grafos – Matrizes de Adjacências

- Uma matriz de adjacências A de um grafo com n vértices é uma matriz $n \times n$ de bits, em que:

Grafos – Matrizes de Adjacências

- Uma matriz de adjacências A de um grafo com n vértices é uma matriz $n \times n$ de bits, em que:
  - $A[i,j] = 1$ se houver uma aresta indo do vértice i para o vértice j no grafo.
  - $A[i,j] = 0$ se não houver aresta indo de i para j

Grafos – Matrizes de Adjacências



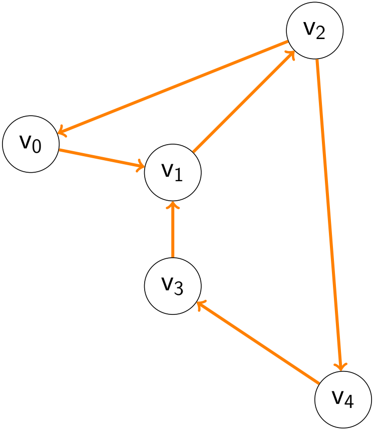
	v_0	v_1	v_2	v_3	v_4
v_0	0	1	1	0	0
v_1	1	0	1	1	0
v_2	1	1	0	0	1
v_3	0	1	0	0	1
v_4	0	0	1	1	0

Grafos – Matrizes de Adjacências

- E se o grafo for dirigido?

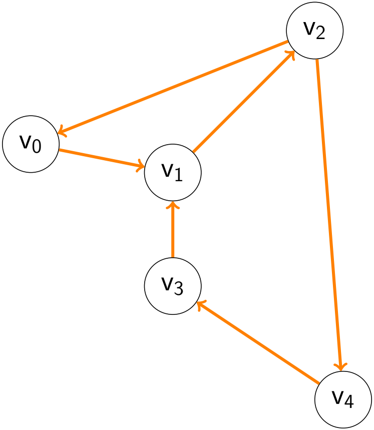
Grafos – Matrizes de Adjacências

- E se o grafo for dirigido?



Grafos – Matrizes de Adjacências

- E se o grafo for dirigido?



A matriz não será simétrica.

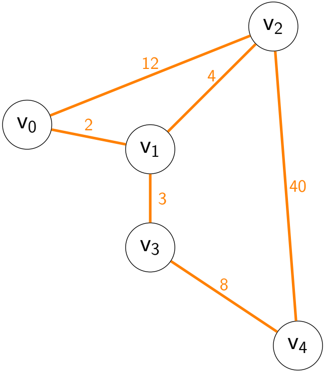
	v_0	v_1	v_2	v_3	v_4
v_0	0	1	0	0	0
v_1	0	0	1	0	0
v_2	1	0	0	0	1
v_3	0	1	0	0	0
v_4	0	0	0	1	0

Grafos – Matrizes de Adjacências

- E se o grafo for ponderado?

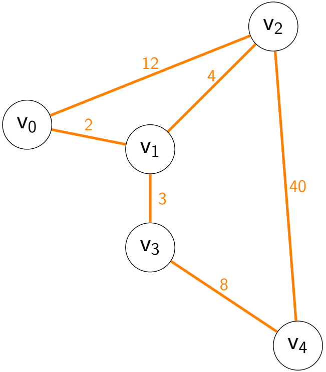
Grafos – Matrizes de Adjacências

- E se o grafo for ponderado?



Grafos – Matrizes de Adjacências

- E se o grafo for ponderado? Não usamos bits.



	v_0	v_1	v_2	v_3	v_4
v_0	0	2	12	0	0
v_1	2	0	4	3	0
v_2	12	4	0	0	40
v_3	0	3	0	0	8
v_4	0	0	40	8	0

Grafos – Matrizes de Adjacências

- E se o grafo for ponderado? Não usamos bits.

- Cuidado!

	v_0	v_1	v_2	v_3	v_4
v_0	0	2	12	0	0
v_1	2	0	4	3	0
v_2	12	4	0	0	40
v_3	0	3	0	0	8
v_4	0	0	40	8	0

Grafos – Matrizes de Adjacências

- E se o grafo for ponderado? Não usamos bits.

- <u>Cuidado!</u>
- Como diferenciar não aresta de um valor válido (como 0)?

	v_0	v_1	v_2	v_3	v_4
v_0	<u>0</u>	2	12	<u>0</u>	<u>0</u>
v_1	2	<u>0</u>	4	3	<u>0</u>
v_2	12	4	<u>0</u>	<u>0</u>	40
v_3	<u>0</u>	3	<u>0</u>	<u>0</u>	8
v_4	<u>0</u>	<u>0</u>	40	8	<u>0</u>

Grafos – Matrizes de Adjacências

- E se o grafo for ponderado? Não usamos bits.

- Cuidado!
- Como diferenciar não aresta de um valor válido (como 0)?
- Deve-se definir um padrão

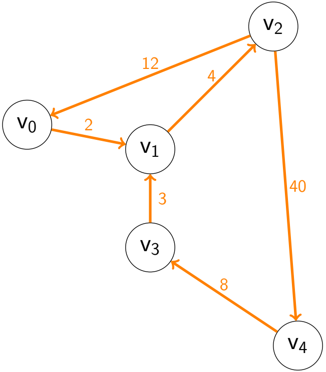
	v_0	v_1	v_2	v_3	v_4
v_0	0	2	12	0	0
v_1	2	0	4	3	0
v_2	12	4	0	0	40
v_3	0	3	0	0	8
v_4	0	0	40	8	0

Grafos – Matrizes de Adjacências

- E se o grafo for dirigido?

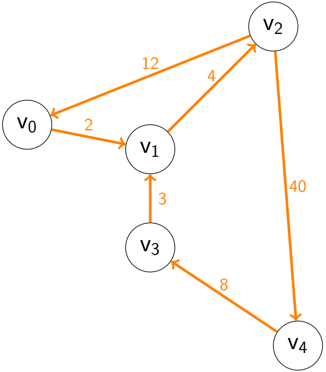
Grafos – Matrizes de Adjacências

- E se o grafo for dirigido?



Grafos – Matrizes de Adjacências

- E se o grafo for dirigido?



	v_0	v_1	v_2	v_3	v_4
v_0	0	2	0	0	0
v_1	0	0	4	0	0
v_2	12	0	0	0	40
v_3	0	3	0	0	0
v_4	0	0	0	8	0

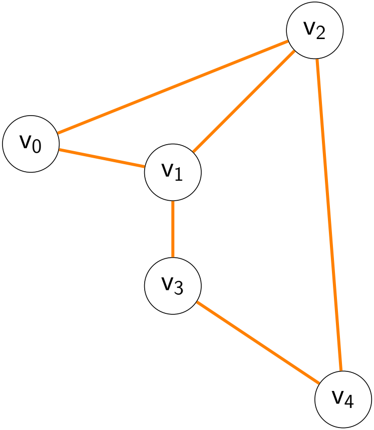
Grafos – Listas de Adjacências

- Uma lista de adjacências de um grafo com n vértices consiste de um arranjo de n listas ligadas, uma para cada vértice no grafo.

Grafos – Listas de Adjacências

- Uma lista de adjacências de um grafo com n vértices consiste de um arranjo de n listas ligadas, uma para cada vértice no grafo.
  - Para cada vértice u, a lista contém todos os vizinhos de u
  - Ou seja, todos os vértices v_i para os quais existe uma aresta (u, v_i)

Grafos – Listas de Adjacências



$v_0 \rightarrow v_1 \rightarrow v_2$

$v_1 \rightarrow v_0 \rightarrow v_2 \rightarrow v_3$

$v_2 \rightarrow v_0 \rightarrow v_1 \rightarrow v_4$

$v_3 \rightarrow v_1 \rightarrow v_4$

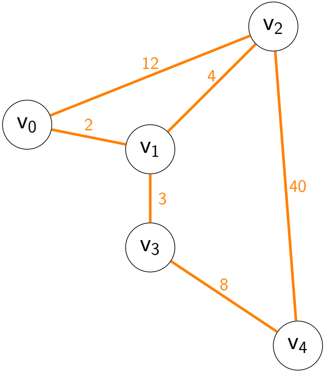
$v_4 \rightarrow v_2 \rightarrow v_3$

Grafos – Listas de Adjacências

- E se o grafo for ponderado?

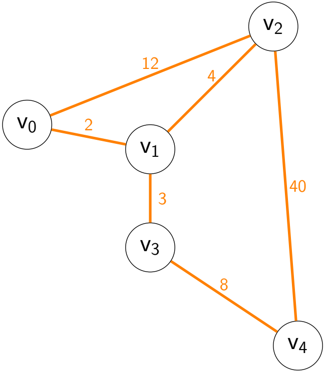
Grafos – Listas de Adjacências

- E se o grafo for ponderado?



Grafos – Listas de Adjacências

- E se o grafo for ponderado? Armazenamos os pesos nas lista.



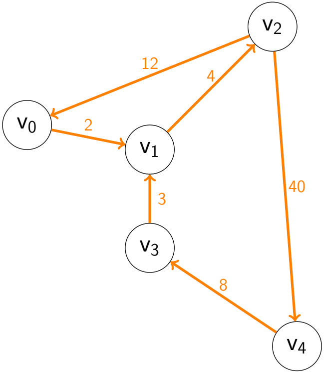
v_0	v_1, 2	v_2, 12	
v_1	v_0, 2	v_2, 4	v_3, 3
v_2	v_0, 12	v_1, 4	v_4, 40
v_3	v_1, 3	v_4, 8	
v_4	v_2, 40	v_3, 8	

Grafos – Listas de Adjacências

- E se o grafo for dirigido?

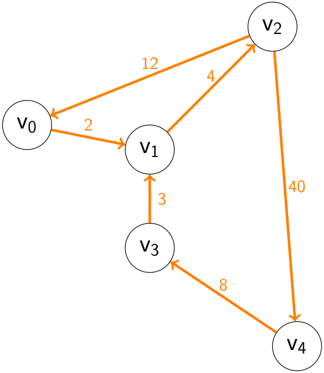
Grafos – Listas de Adjacências

- E se o grafo for dirigido?



Grafos – Listas de Adjacências

- E se o grafo for dirigido?



$v_0 \rightarrow v_1,\ 2$

$v_1 \rightarrow v_2,\ 4$

$v_2 \rightarrow v_0,\ 12 \rightarrow v_4,\ 40$

$v_3 \rightarrow v_1,\ 3$

$v_4 \rightarrow v_3,\ 8$

Grafos – Representação

- E quando usamos uma ou outra representação?

Grafos – Representação

- E quando usamos uma ou outra representação?
- Vai depender da densidade do grafo
  - Se denso (quando possui muitas arestas em relação ao número de vértices) ou esparso (com relativamente poucas arestas)
- Vai depender das operações que queremos executar

Grafos – Representação

Nas próximas aulas implementaremos as estruturas de dados e os respectivos algoritmos, utilizando a linguagem C, para gerenciarmos grafos com base nestas duas representações.

Referências

1. AHO, A. V.; HOPCROFT, J. E.; ULLMAN, J. D. Data Structure and Algorithms. Readings, Addison Wesley, 1983.
2. CORMEN, H. T.; LEISERSON, C.E.; RIVEST, R.L. Introduction to Algorithms, MIT Press, McGraw-Hill, 1999.

Algoritmos e Estruturas de Dados II
Aula 02 – Grafos: Conceitos Básicos

Prof. Luciano A. Digiampietri
digiampietri@usp.br
@digiampietri