# AULA 1

# Conceitos básicos e representação de grafos

**Karina Valdivia Delgado**

# Roteiro

Motivação
Conceitos básicos
Representação

# Motivação

Um grafo é uma abstração que permite codificar relacionamentos entre pares de objetos (definição informal)

Em que:

Os objetos são os **vértices** do grafo

Os relacionamentos são as **arestas** do grafo

# Motivação

**Objeto:** cidades

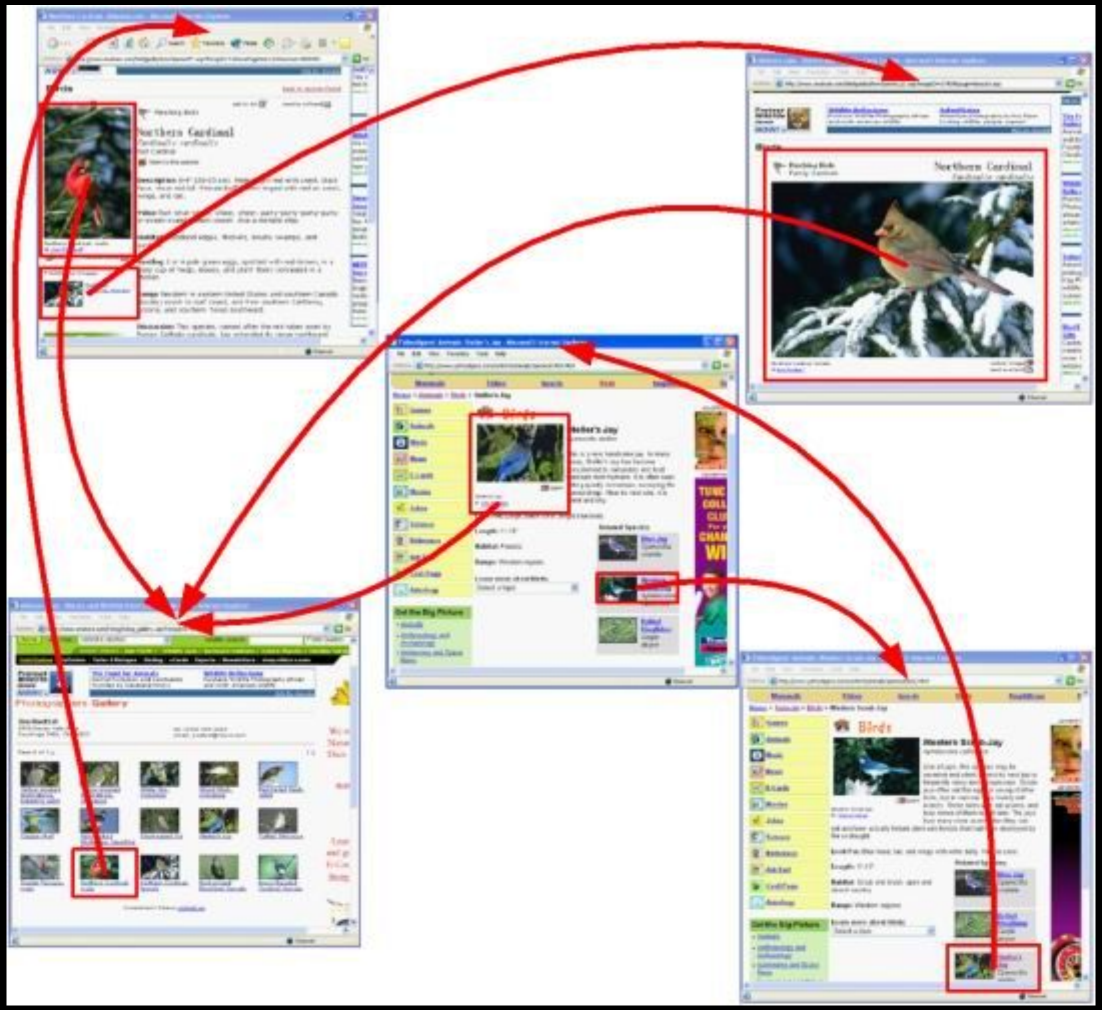
**Relacionamento:** voo comercial entre duas cidades

Fonte: Wikipedia

# Motivação

**Objeto:** página web

**Relacionamento:** link de uma página para outra



# Motivação

**Existe um tipo abstrato de dados (TAD=conjunto de operações associado a uma estrutura de dados) usado para modelar tais situações!**

**grafo**

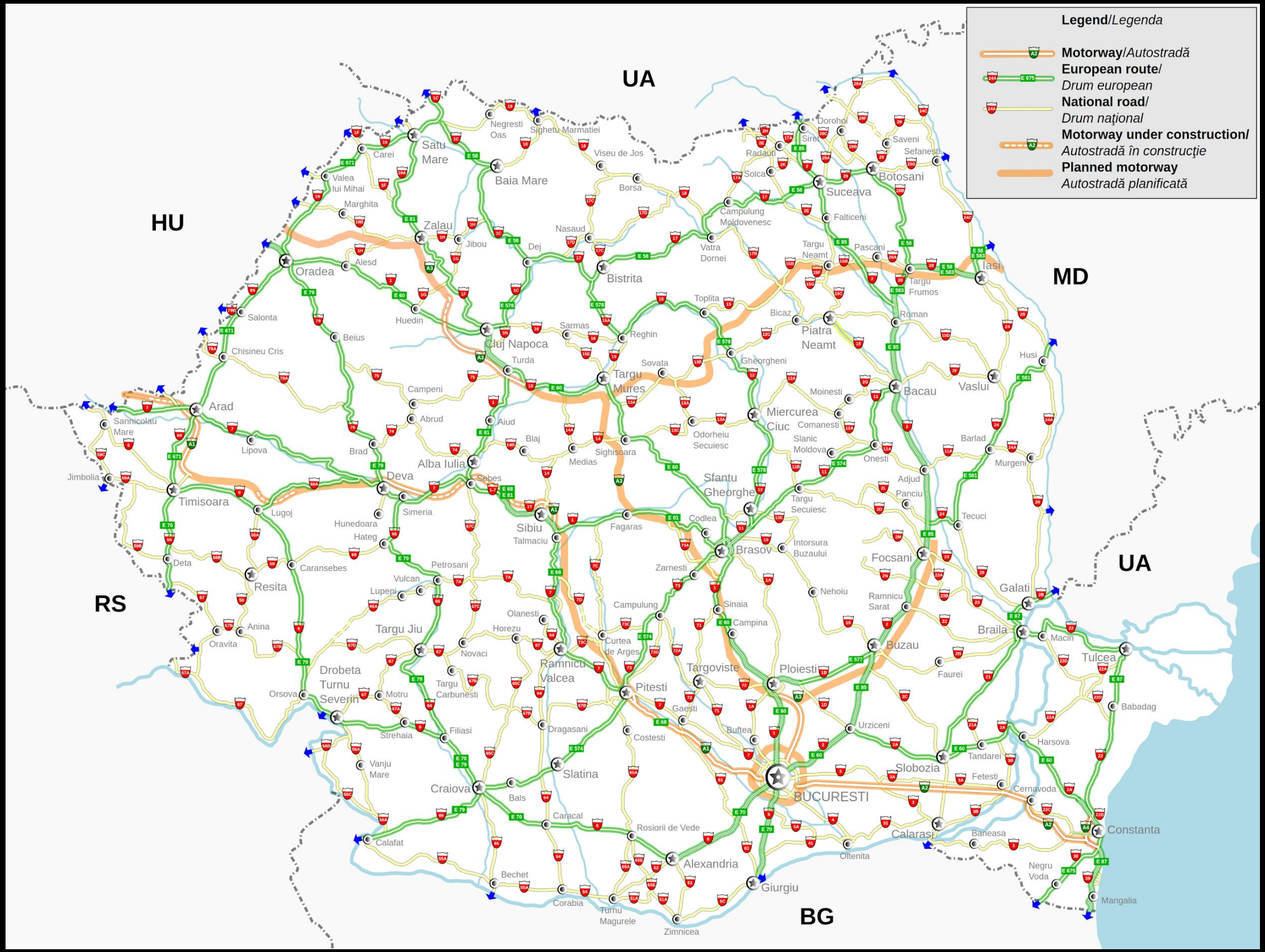
**Muitos problemas podem ser resolvidos com o mesmo algoritmo em cima da abstração**

# Motivação

## Quantos caminhos existem para ir da cidade X até a cidade Y?

## Qual é o menor(melhor) caminho entre X e Y?

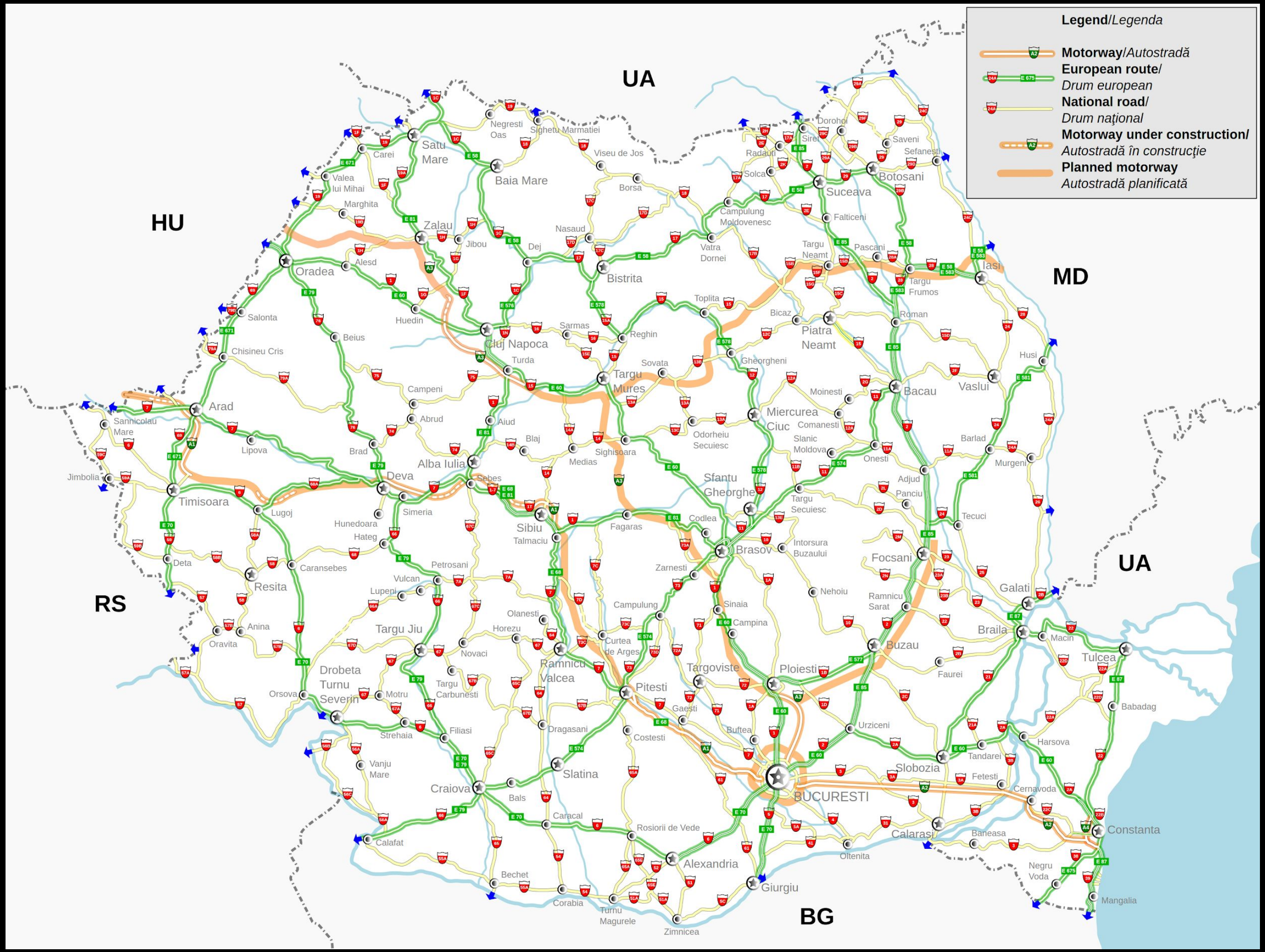
## Existe um caminho para ir de uma cidade a outra?




Fonte: Wikipedia

# Motivação

**Vértices:** cidades
**Arestas:** estradas entre as cidades




Fonte: Wikipedia

# Motivação

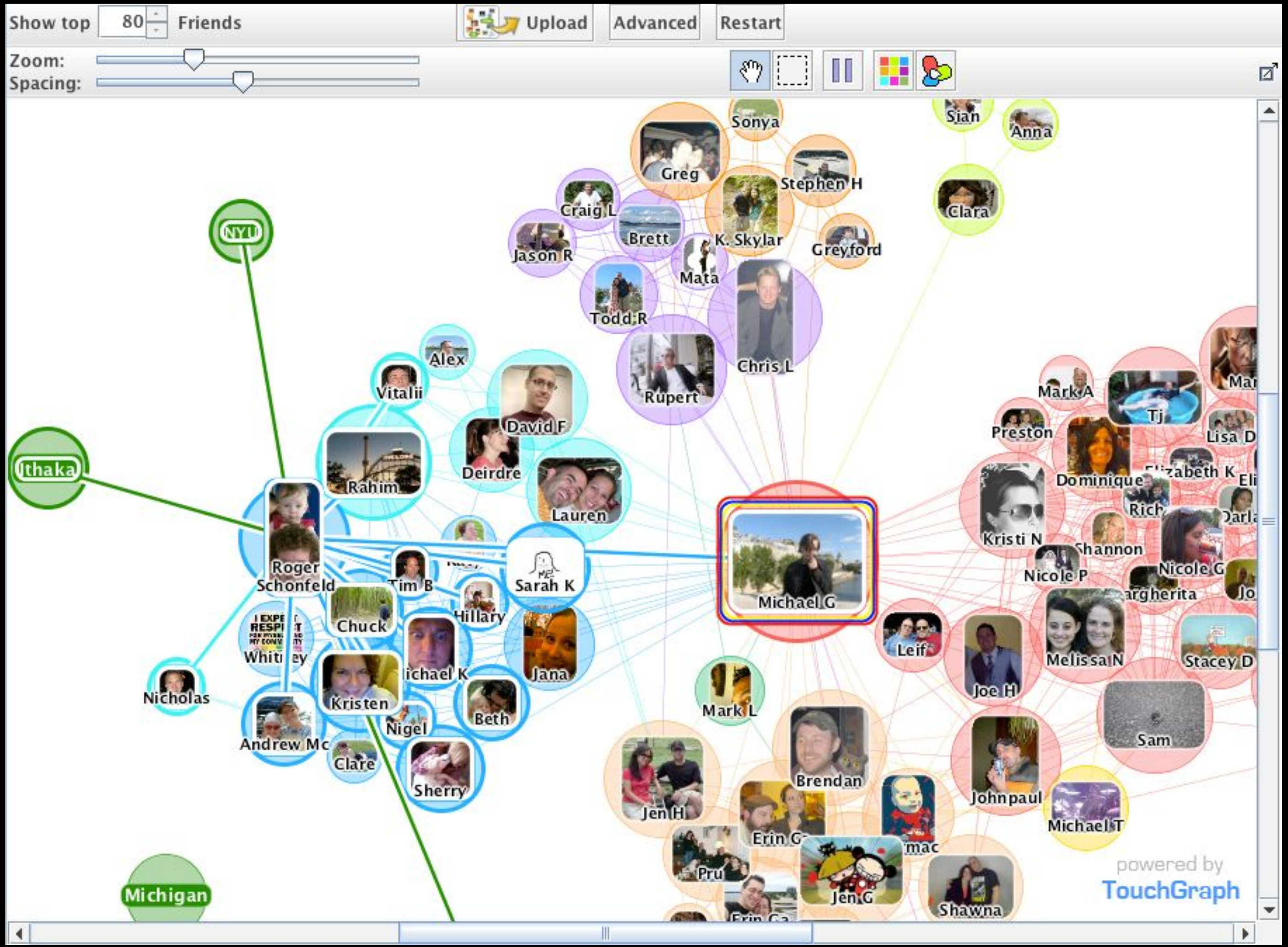
- Temos N professores e M disciplinas
- Dado o que cada professor pode lecionar, é possível que as M disciplinas sejam oferecidas simultaneamente?
- Qual o maior número de disciplinas que podem ser oferecidas?

**Vértices: professoresse e disciplinas**
**Arestas: disciplinas que cada professor pode lecionar**

# Motivação

**Como saber se duas pessoas estão conectadas através de uma sequência de relacionamentos (interligadas via relacionamentos declarados)?**

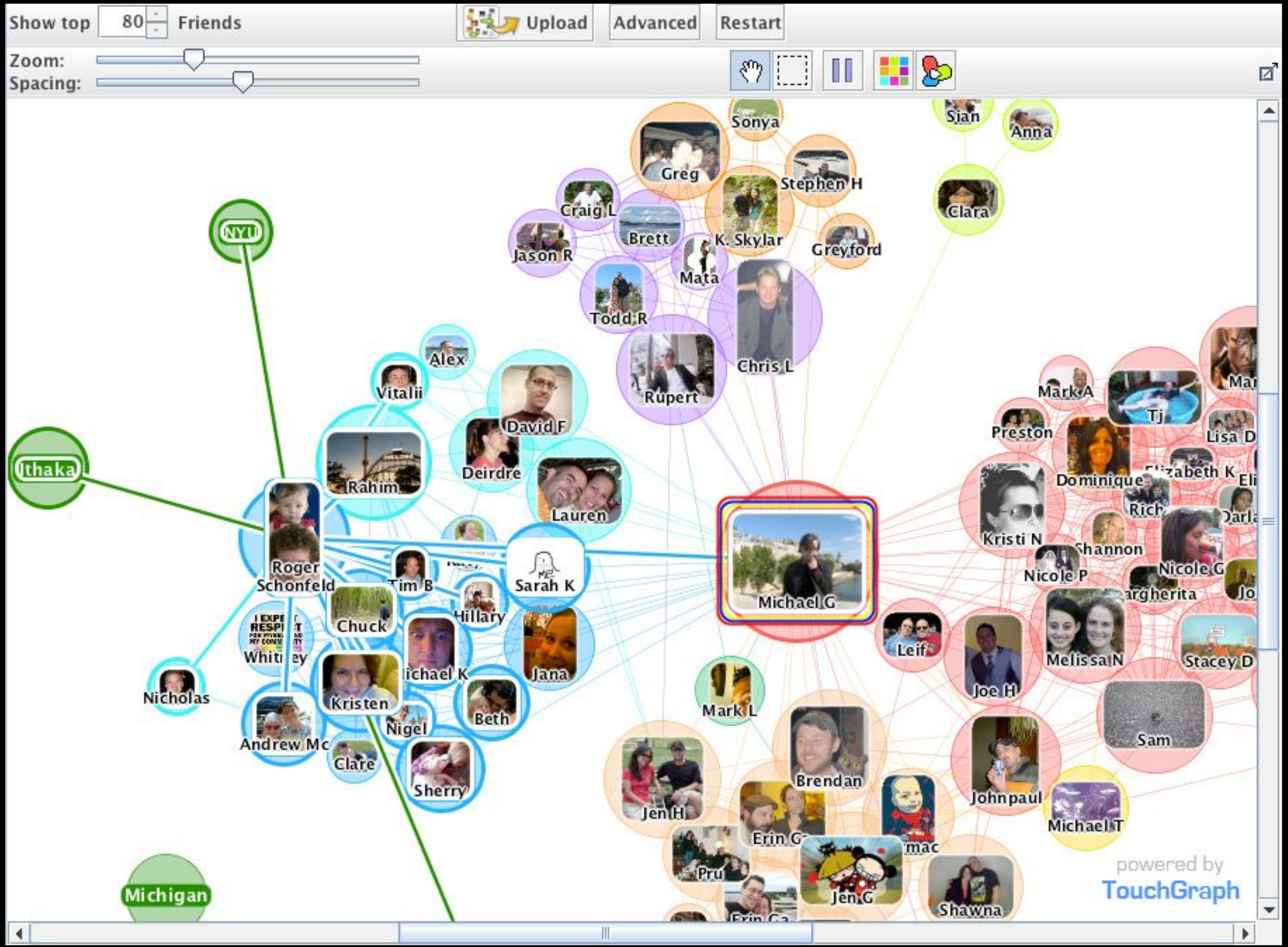
**Qual é o menor caminho entre duas pessoas?**




**Fonte: Flickr**

# Motivação

**Vértices:** pessoas
**Arestas:** relacionamentos declarados

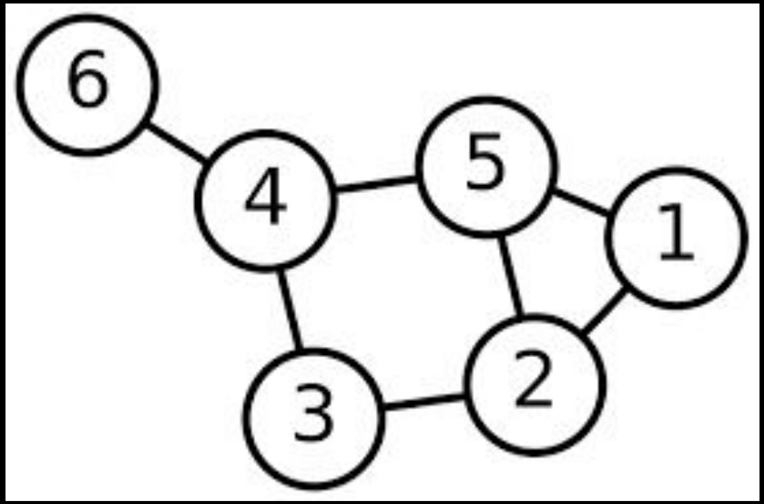



Fonte: Flickr

# Definição de grafo

**Grafo:** conjunto de vértices e arestas.

- **Vértice:** objeto simples que pode ter nome e outros atributos.
- **Aresta:** conexão entre dois vértices.




Fonte: Wikipedia

# Definição de grafo

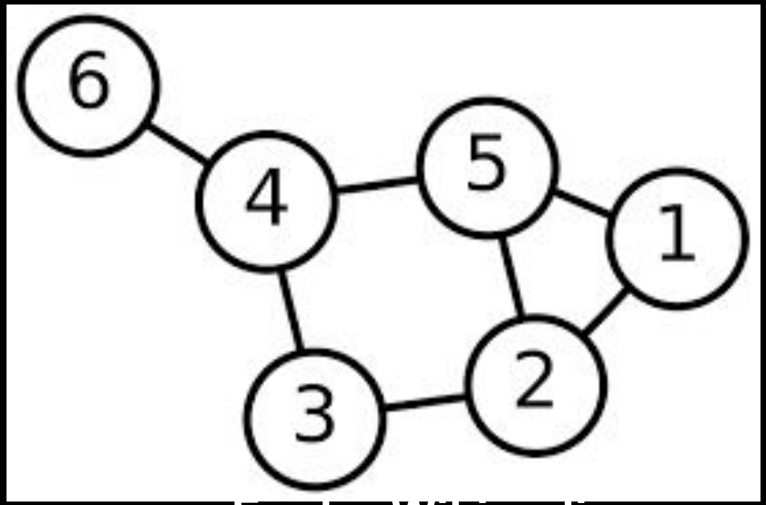
Formalmente um grafo G é definido por

$$G=(V,A)$$

Em que:

– V é o conjunto de vértices

– A é o conjunto de arestas

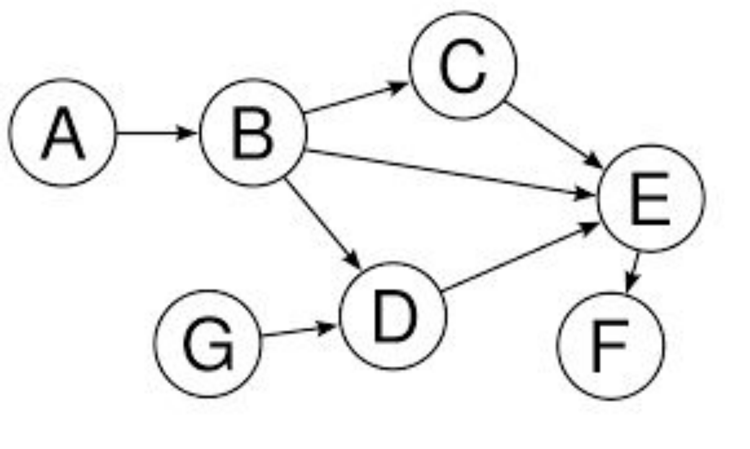



Fonte: Wikipedia

# Definição de grafo direcionado

Um **grafo direcionado** G é um par (V,A), em que:

– V é um conjunto finito de vértices
– A é uma relação binária em V (i.e., uma aresta é um par **ordenado** de vértices)

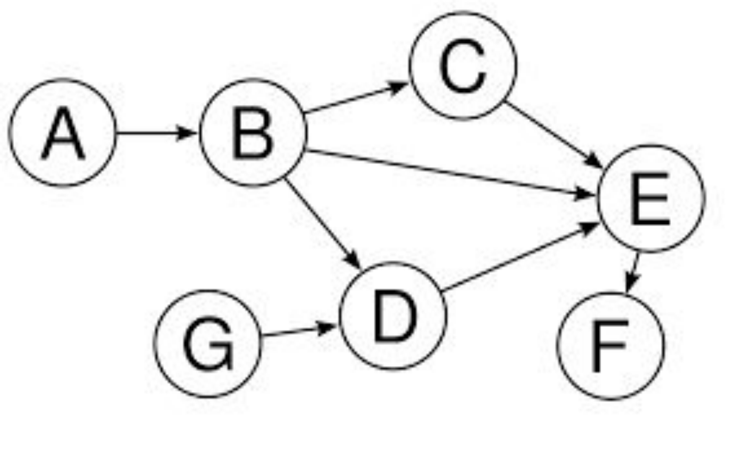



Fonte: Wikipedia

# Definição de grafo direcionado

**Um grafo direcionado G é um par (V,A), em que:**

**– V é um conjunto finito de vértices**
**– A é uma relação binária em V (i.e., uma aresta é um par ordenado de vértices)**



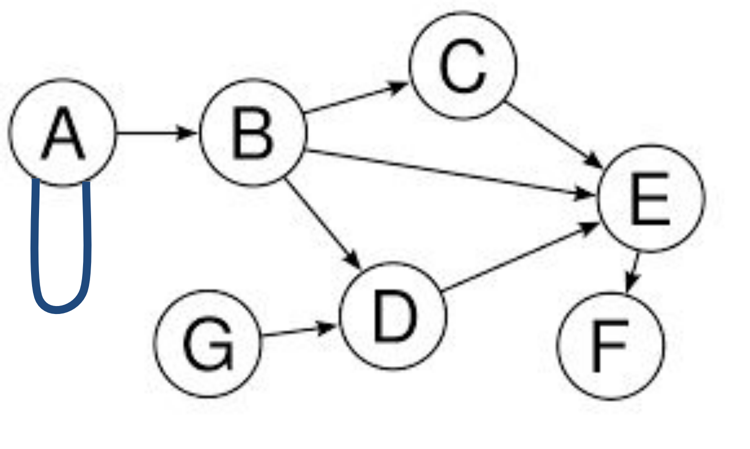

**Fonte: Wikipedia**

A aresta (B, E) sai do vértice B e entra no vértice E . Dizemos que o vértice E é adjacente ao vértice B.

# Definição de grafo direcionado

**Um grafo direcionado G é um par (V,A), em que:**

**– V é um conjunto finito de vértices**
**– A é uma relação binária em V (i.e., uma aresta é um par ordenado de vértices)**

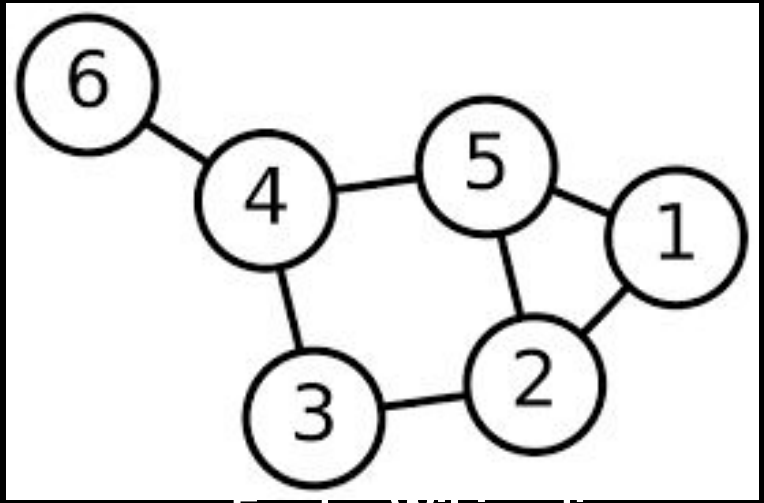



**Fonte: Wikipedia**

Podem existir arestas de um vértice para ele mesmo, chamadas de self-loops.

# Definição de grafo não direcionado

Um grafo não direcionado G é um par (V, A), em que:
– O conjunto de arestas A é constituído de pares de vértices não ordenados.
– As arestas (u, v) e (v , u) são consideradas como uma única aresta.



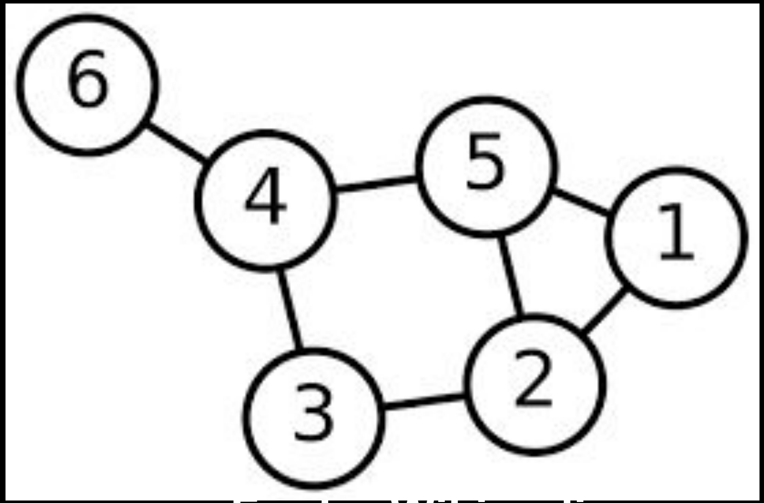

Fonte: Wikipedia

# Definição de grafo não direcionado

Um **grafo não direcionado** G é um par (V, A), em que:

– O conjunto de arestas A é constituído de pares de vértices não ordenados.

– As arestas (u, v) e (v , u) são consideradas como uma única aresta.

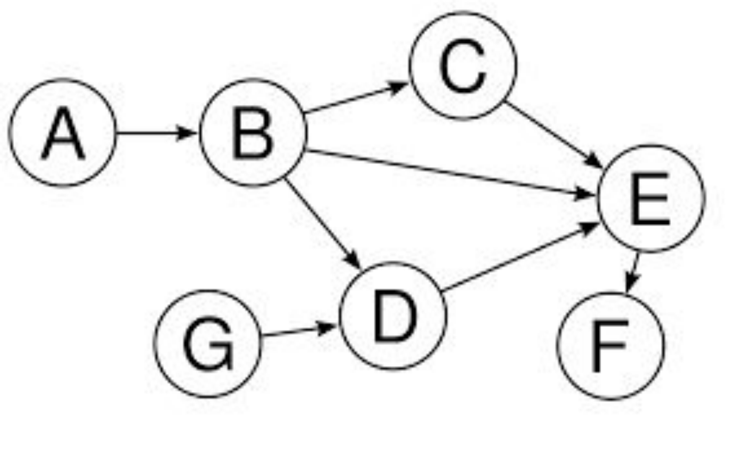


Fonte: Wikipedia

A relação de adjacência é simétrica.
Self-loops não são permitidos.

# Grau de um vértice em grafos direcionados

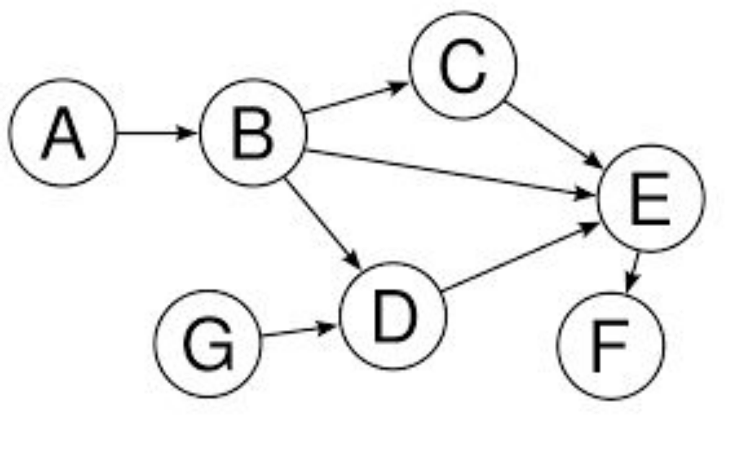
**Grau de saída:** número de arestas que saem do vértice.
**Grau de entrada:** número de arestas que chegam no vértice.
**Grau de um vértice:** grau de saída + grau de entrada.




Fonte: Wikipedia

# Grau de um vértice em grafos direcionados

**Grau de saída:** número de arestas que saem do vértice.
**Grau de entrada:** número de arestas que chegam no vértice.
**Grau de um vértice:** grau de saída + grau de entrada.

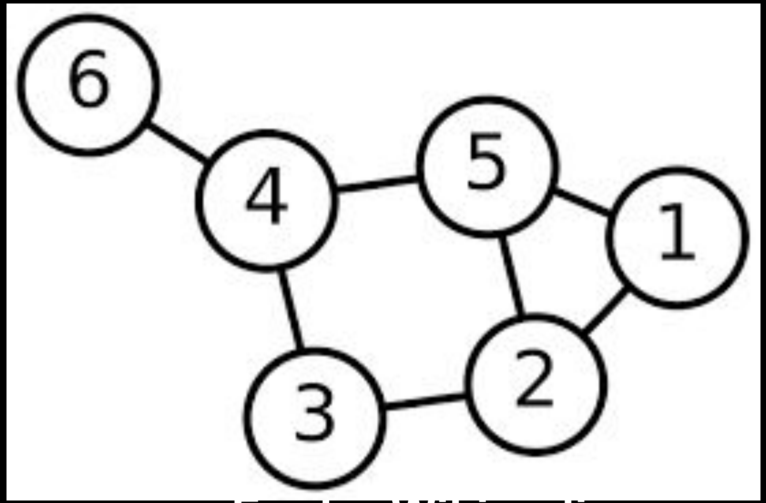



Fonte: Wikipedia

O vértice B tem grau de saída 3, grau de entrada 1 e grau 4.

# Grau de um vértice em grafos não direcionados

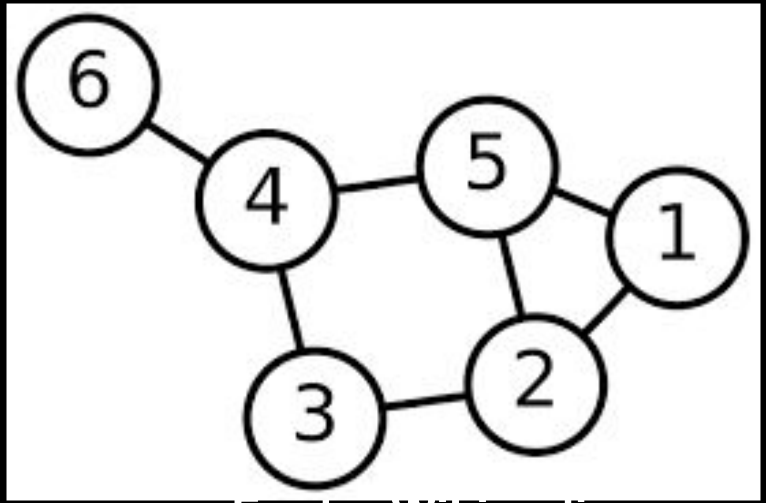
O grau de um vértice é o número de arestas que incidem nele.




Fonte: Wikipedia

# Grau de um vértice em grafos não direcionados

**O grau de um vértice é o número de arestas que incidem nele.**

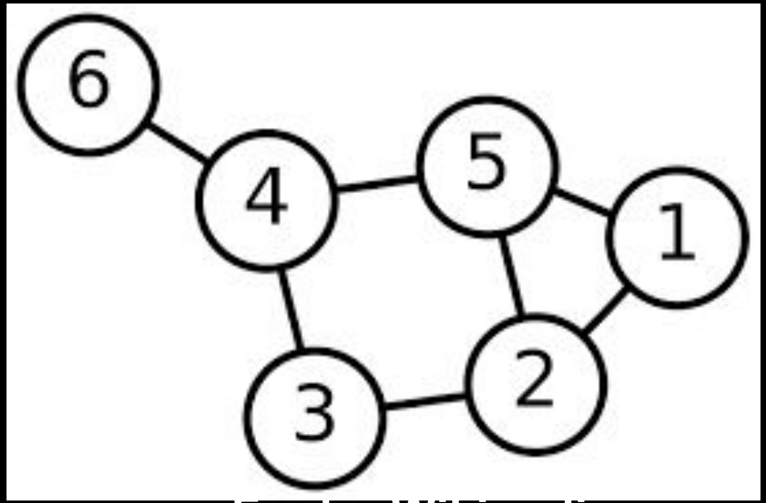



Fonte: Wikipedia

**Um vértice de grau zero é dito isolado ou não conectado.**

# Grau de um vértice em grafos não direcionados

O grau de um vértice é o número de arestas que incidem nele.



Fonte: Wikipedia

O vértice 5 tem grau 3

# Caminho entre vértices

- Um caminho de comprimento k de um vértice x a um vértice y em um grafo G = (V , A) é uma seqüência de vértices

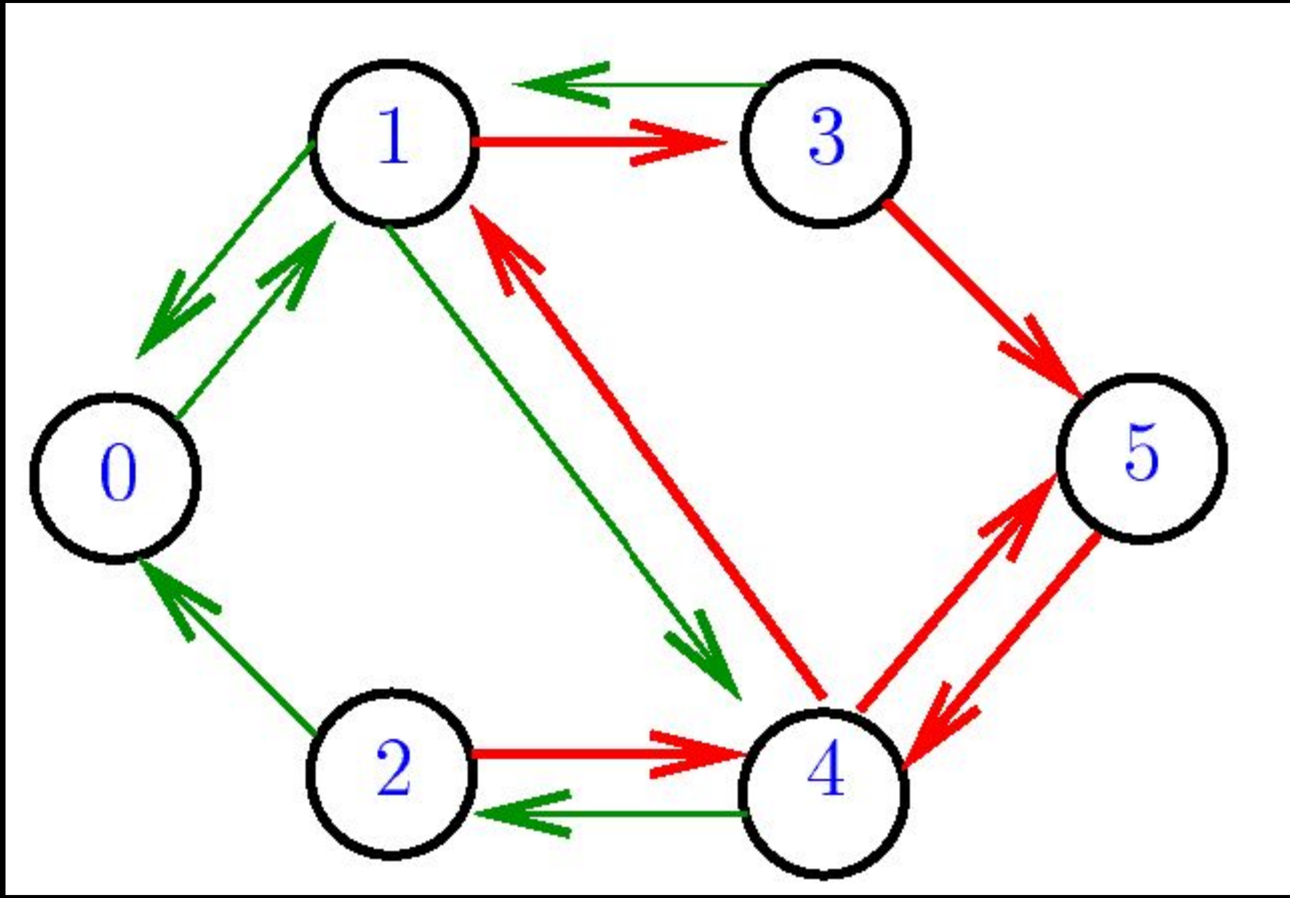
$$(v_0 , v_1 , v_2 , . . . , v_k )$$

tal que $x = v_0$ e $y = v_k$ , e $(v_{i-1} , v_i ) \in A$ para $i = 1, 2, . . . , k$ .

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

# Caminho entre vértices

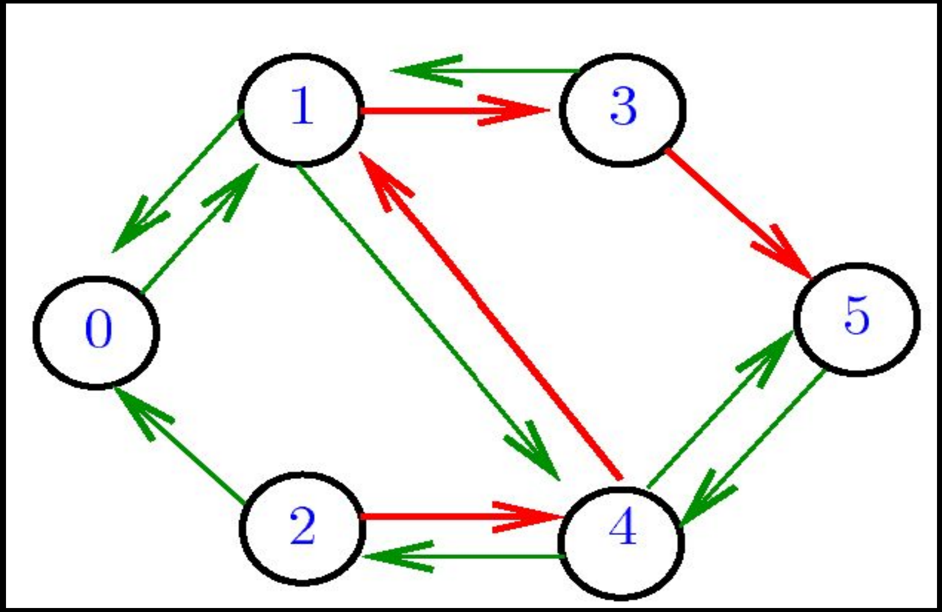
- (2,4,1,3,5,4,5) é um caminho do vértice 2 até o vértice 5 de comprimento 6



# Caminho entre vértices

Se existir um caminho c de x a y então y é alcançável a partir de x via c.

- Um caminho é simples se todos os vértices do caminho são distintos. Exemplo: (2,4,1,3,5)
-



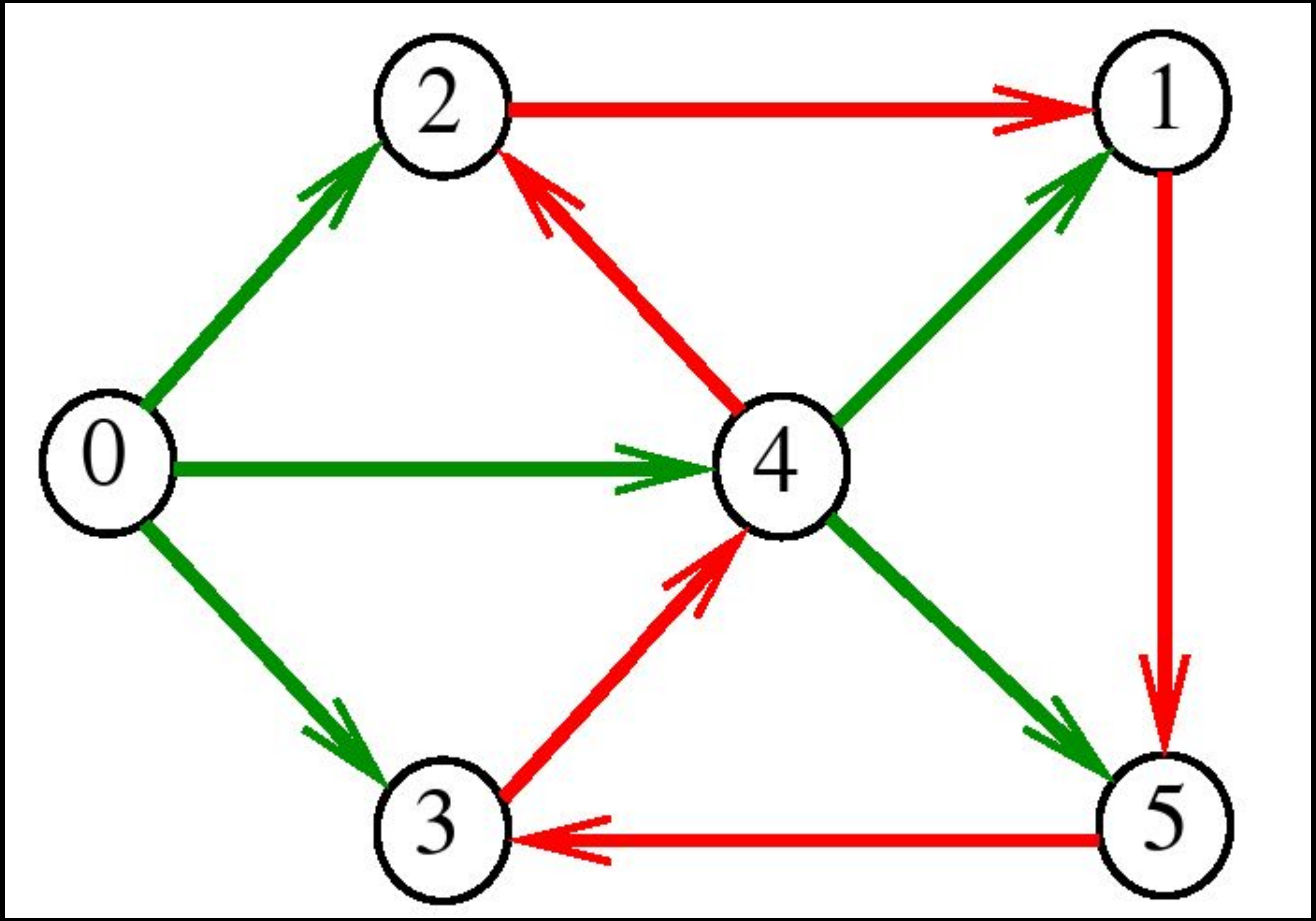
# Ciclos em um grafo direcionado

Um caminho $(v_0, v_1, \ldots, v_k)$ forma um ciclo se $v_0 = v_k$ e o caminho contém pelo menos uma aresta.

- – O ciclo é simples se os vértices $v_1, v_2, \ldots, v_k$ são distintos.
- – O self-loop é um ciclo de tamanho 1.

# Ciclos em um grafo direcionado

Exemplo: (2,1,5,3,4,2) é um ciclo simples

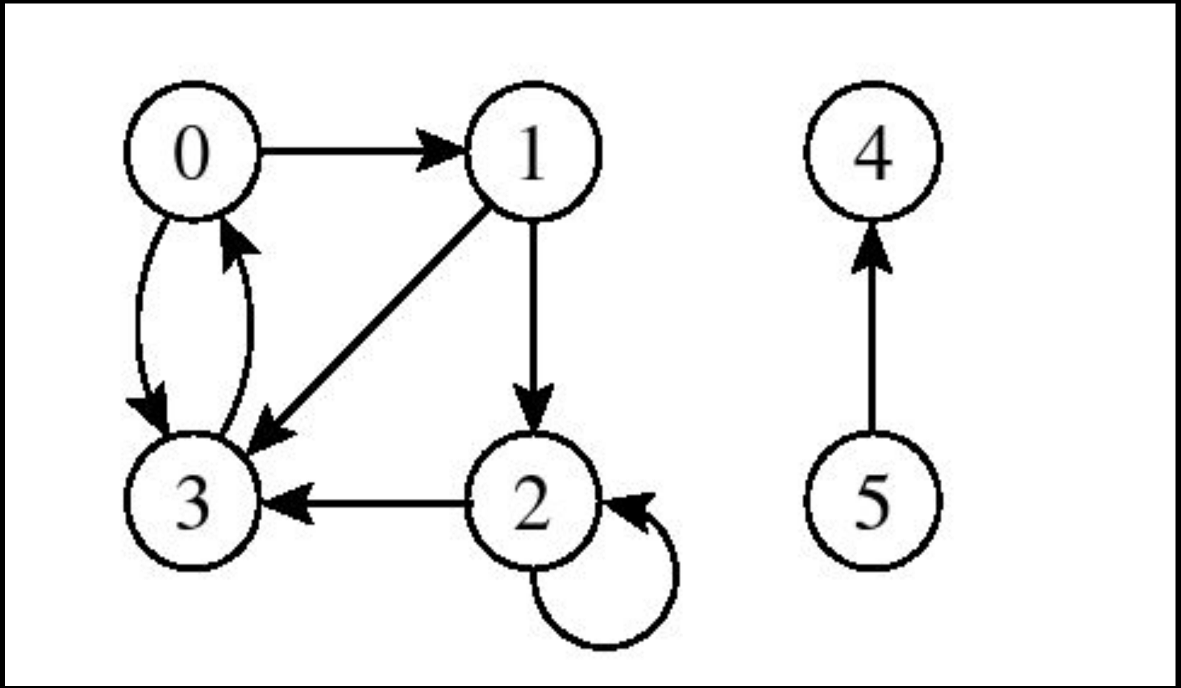


# Ciclos em um grafo direcionado

- Dois caminhos $(v_0, v_1, \ldots, v_k)$ e $(v'_0, v'_1, \ldots, v'_k)$ formam o mesmo ciclo se existir um inteiro j tal que $v'_i = v_{(i+j) \bmod k}$ para $i = 0, 1, \ldots, k - 1$.

# Ciclos em um grafo direcionado

- Exemplo: o caminho (0, 1, 3, 0) forma o mesmo ciclo que os caminhos (1, 3, 0, 1) e (3, 0, 1, 3).



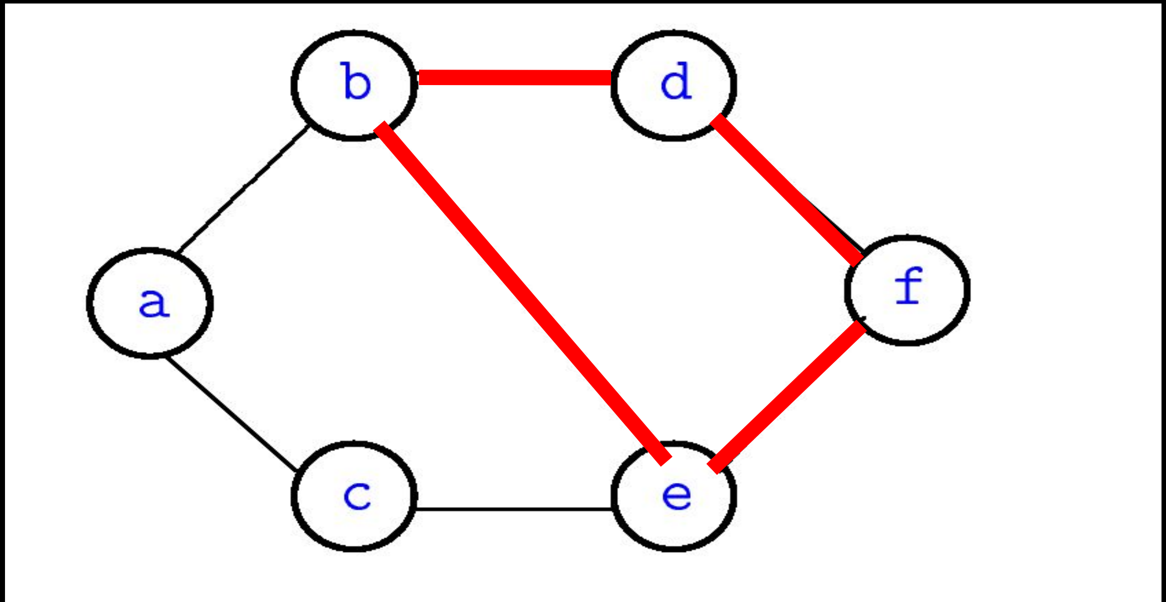
# Ciclos em um grafo não direcionado

- Um caminho $(v_0, v_1, \ldots, v_k)$ forma um ciclo se $v_0 = v_k$ e o caminho contém pelo menos três arestas.
- O ciclo é simples se os vértices

  $v_1, v_2, \ldots, v_k$ são distintos.

# Ciclos em um grafo não direcionado

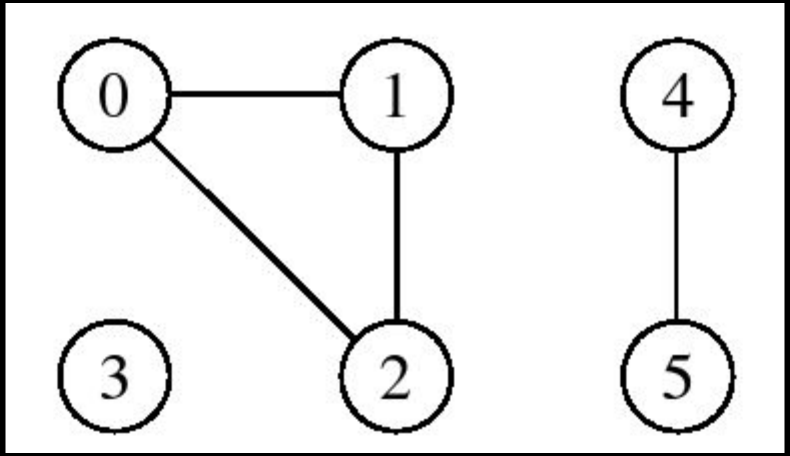
- Exemplo: o caminho (b, d, f, e, b) é um ciclo simples



# Componentes conectados

- Um grafo não direcionado é conectado se  cada par de vértices está conectado por um caminho.
- Os componentes conectados são as porções conectadas de um grafo.
- Um grafo não direcionado é conectado se ele tem exatamente um componente conectado.

# Componentes conectados

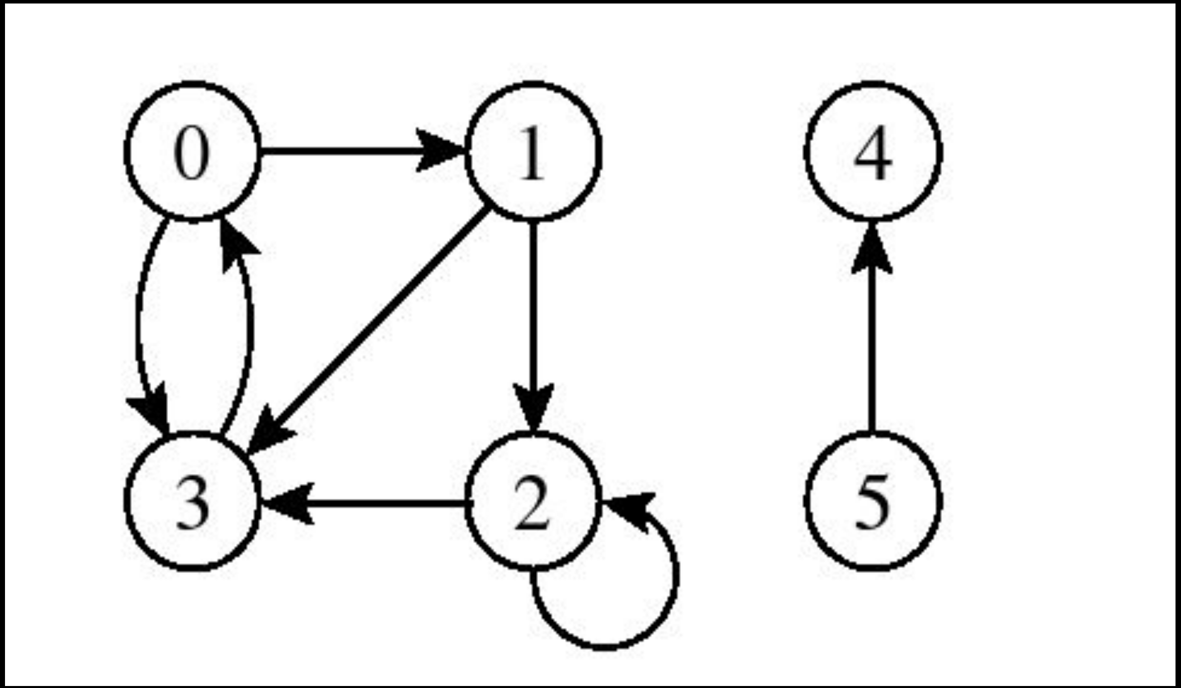
- Os componentes conectados são: {3} {0,1,2} e {4,5} e o grafo não é conectado uma vez que ele tem mais de um componente conectado.
- 

# Componentes fortemente conectados

- Um grafo direcionado G = (V,A) é fortemente conectado se cada dois vértices quaisquer são alcançáveis um a partir do outro.
- Os componentes fortemente conectados de um grafo direcionado são conjuntos de vértices sob a relação “são mutuamente alcançáveis”.
- Um grafo direcionado fortemente conectado tem apenas um componente fortemente conectado

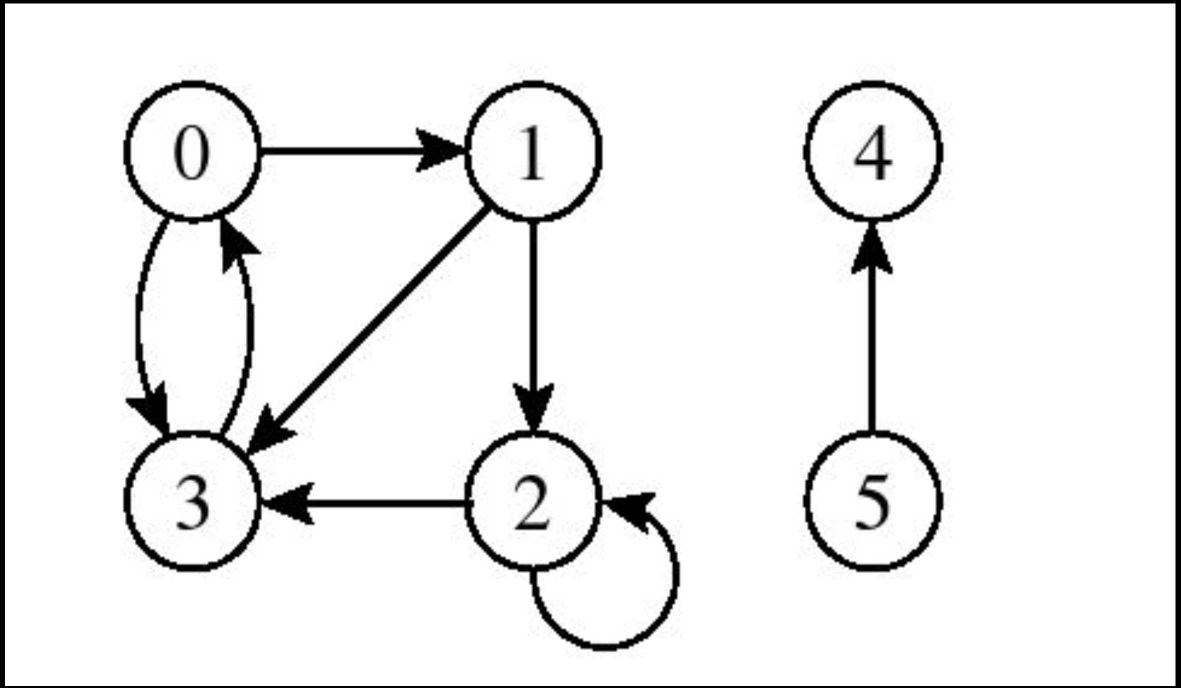
# Componentes fortemente conectados

- quais os componentes fortemente conectados?



# Componentes fortemente conectados

- {0, 1, 2, 3}, {4} e {5} são os componentes fortemente conectados
- {4, 5} não é um componente fortemente conectado

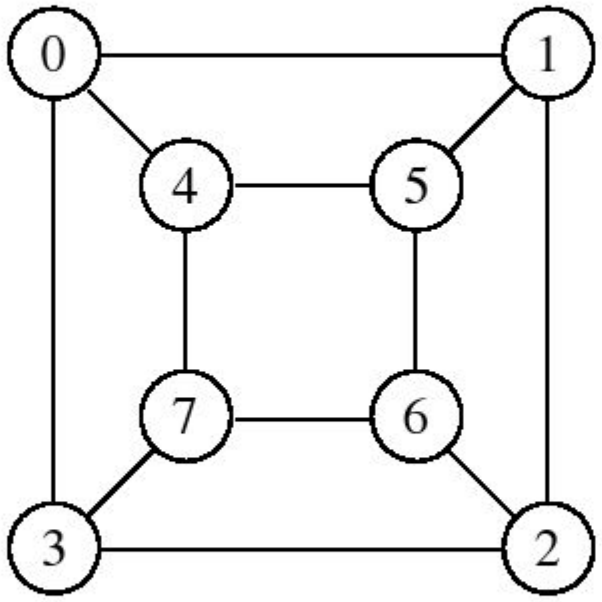


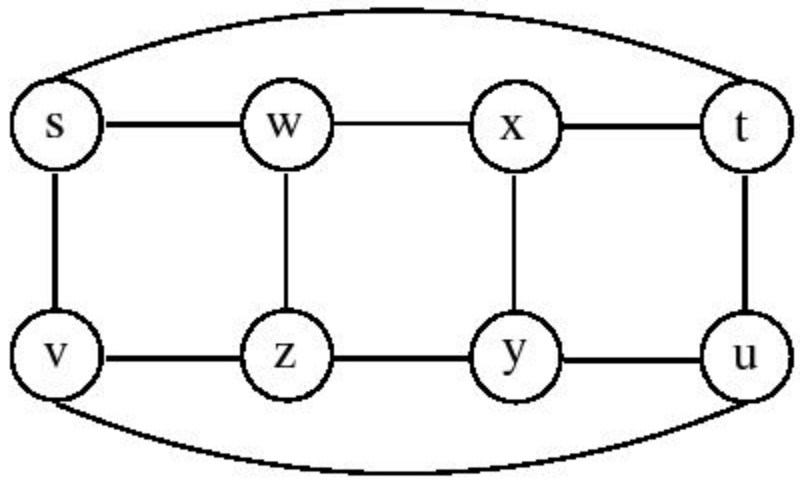
# Grafos Isomorfos

- G = (V , A) e G'= (V', A') são isomorfos se existir uma bijeção $f : V \rightarrow V'$ tal que

  $(u, v) \in A$ se e somente se $(f(u), f(v)) \in A'$ .

É possível re-rotular os vértices de G para serem rótulo de G'?

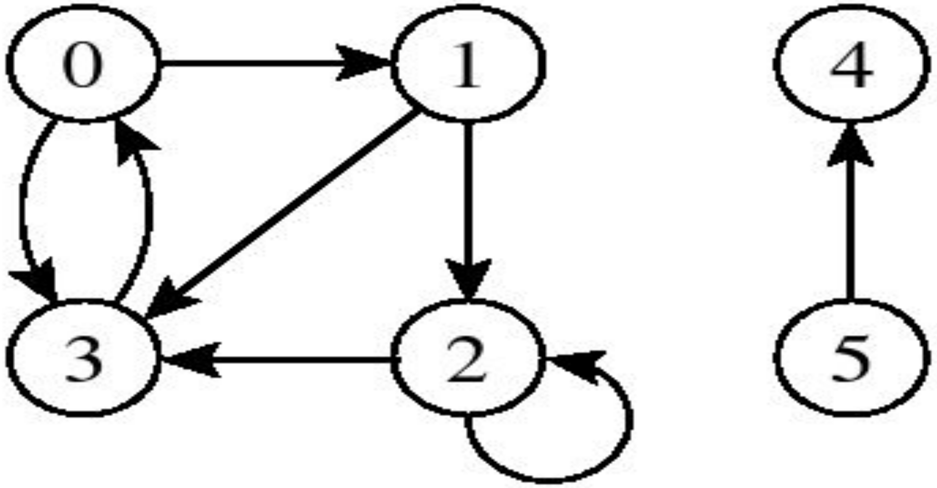


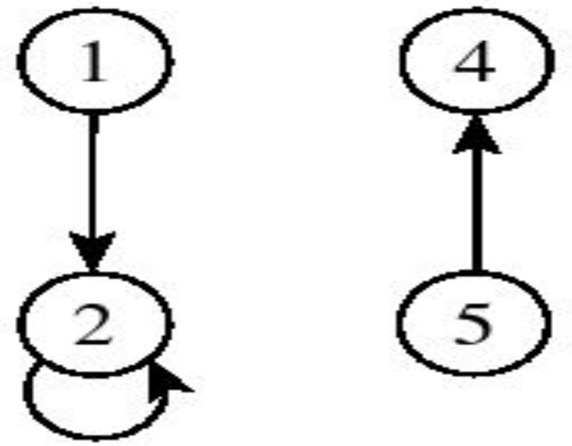
# Subgrafos

- Um grafo $G' = (V', A')$ é um subgrafo de $G = (V, A)$ se $V' \subseteq V$ e $A' \subseteq A$.
- Dado um conjunto $V' \subseteq V$, o subgrafo induzido por $V'$ é o grafo $G' = (V', A')$, em que $A' = \{(u, v) \in A \mid u, v \in V'\}$.

# Subgrafos




- O subgrafo induzido pelo conjunto de vértices V'={1,2,4,5} é:

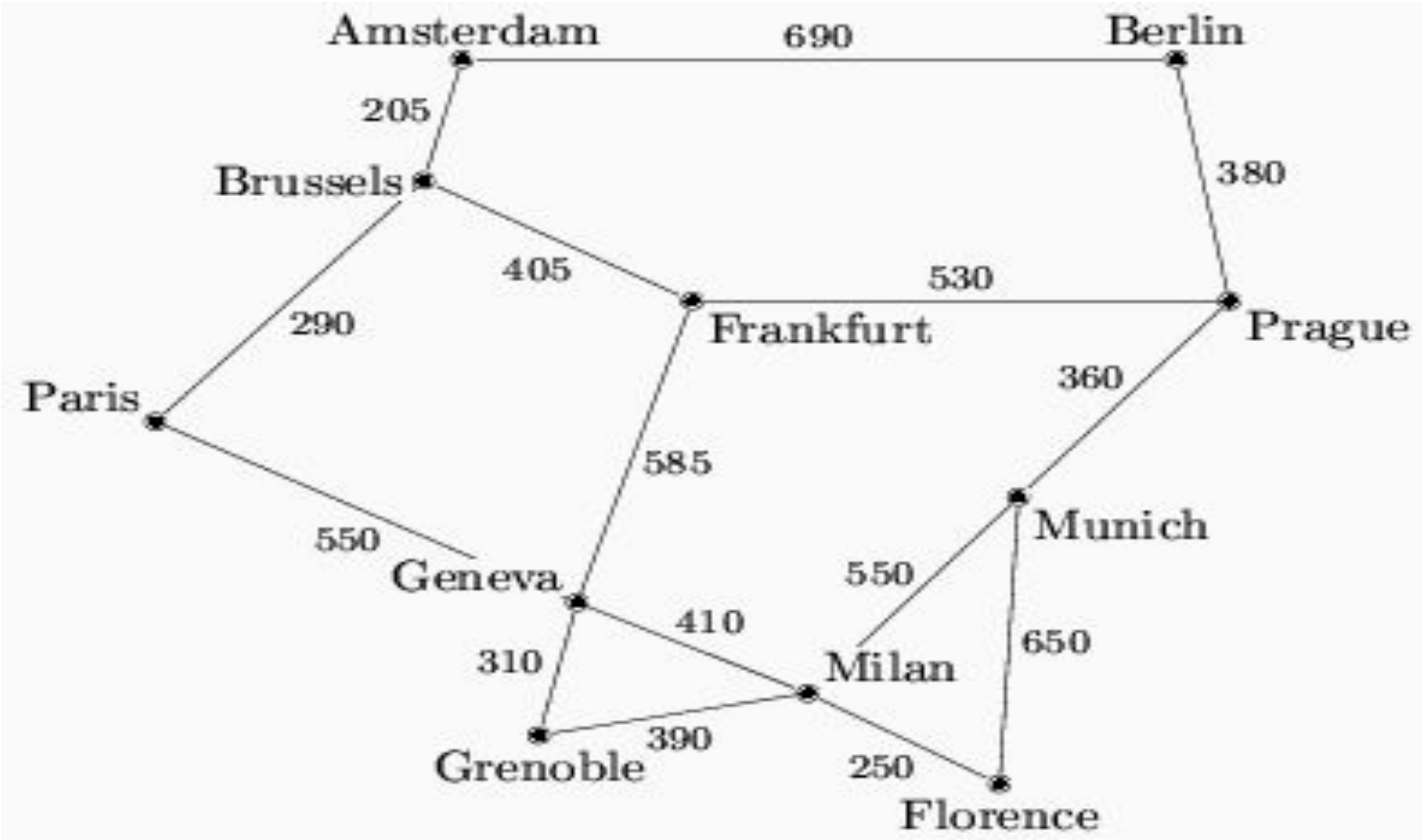


# Outras clasificações de grafos

- **Grafo ponderado:** possui pesos associados às arestas.

# Outras clasificações de grafos

- **Grafo ponderado:** possui pesos associados às arestas.



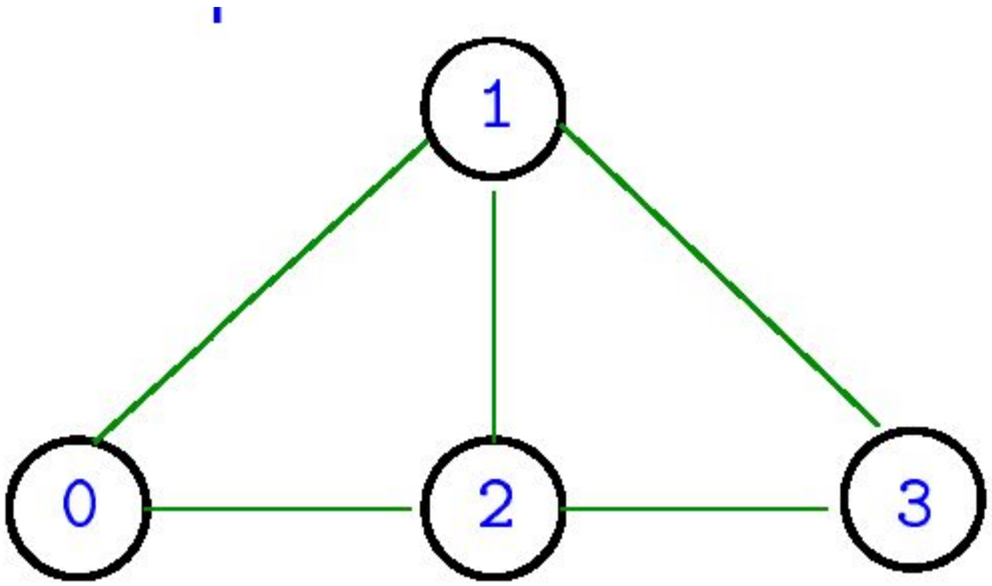
# Outras clasificações de grafos

- **Grafo bipartido:** grafo não direcionado

  G = (V , A) no qual V pode ser particionado em dois conjuntos V1 e V2 tal que (u, v ) $\in$ A implica que u $\in$ V1 e v $\in$ V2 ou u $\in$ V2 e v $\in$ V1 (todas as arestas ligam os dois conjuntos V1 e V2 ).

# Grafos completos

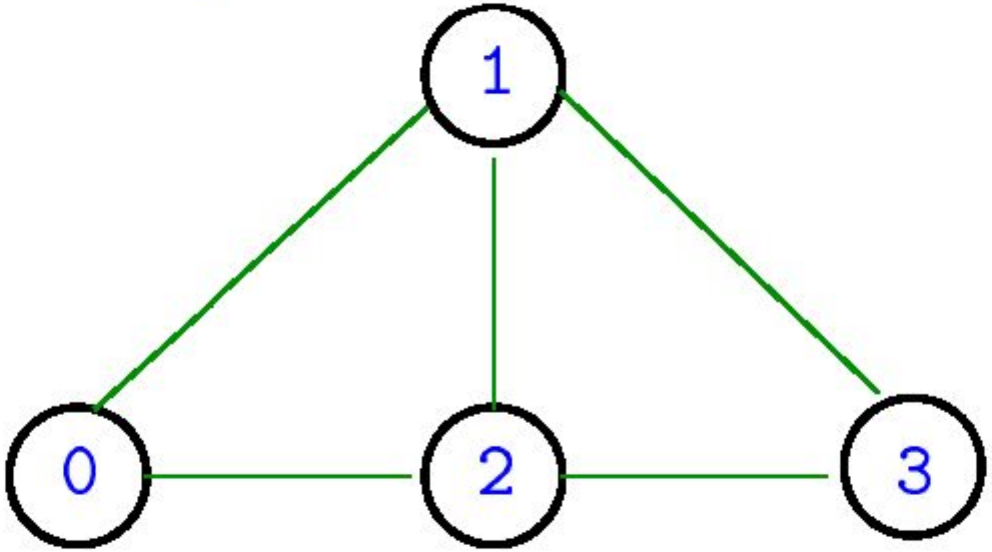
- Um grafo completo é um grafo não direcionado no qual todos os pares de vértices são adjacentes.




É completo?

# Grafos completos

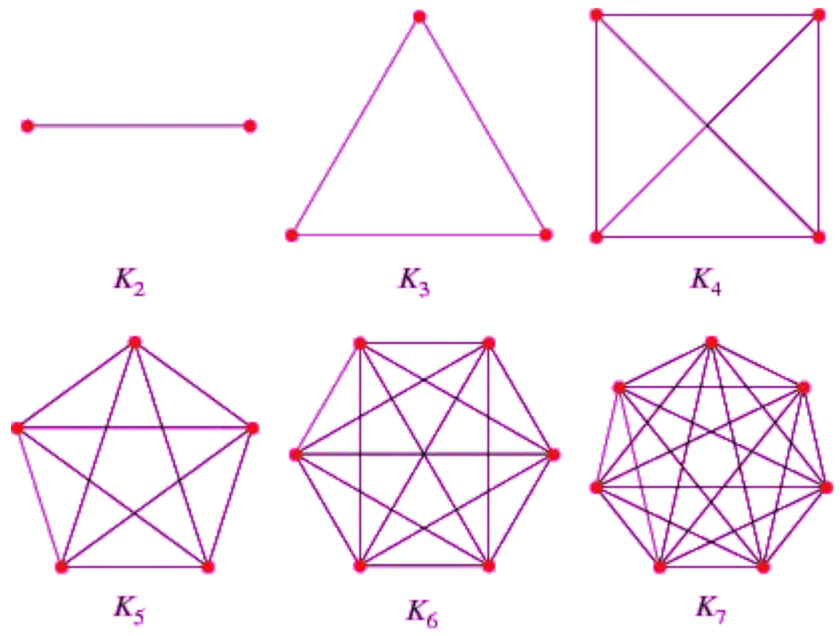
- Um grafo completo é um grafo não direcionado no qual todos os pares de vértices são adjacentes.




É completo?
não

# Grafos completos

- Um grafo completo é um grafo não direcionado no qual todos os pares de vértices são adjacentes.



# Grafos completos

- Quantas arestas possui um grafo completo com V vértices?

# Grafos completos

- Quantas arestas possui um grafo completo com V vértices?
- Possui $(|V|^2 - |V|)/2 = |V|(|V| - 1)/2$ arestas, pois:

  do total de $|V|^2$ pares possíveis de vértices devemos subtrair $|V|$ self-loops e dividir por 2 (cada aresta ligando dois vértices é contada duas vezes no grafo direcionado).

# Grafos completos

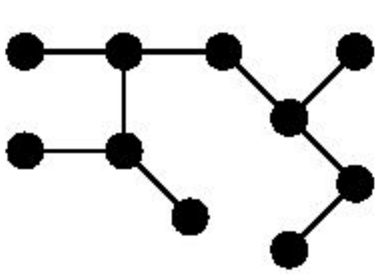
- Qual o número total de grafos diferentes com |V| vértices?

# Grafos completos

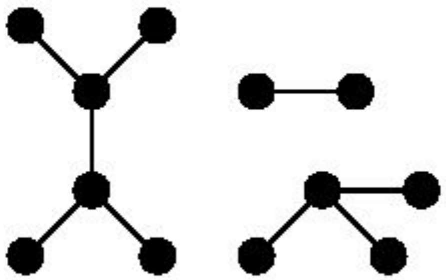
- Qual o número total de grafos diferentes com $|V|$ vértices? Ou seja, qual o número de maneiras diferentes de escolher um subconjunto a partir de $|V|(|V| - 1)/2$ possíveis arestas?

# Árvores e floresta

- Árvore livre: grafo não direcionado acíclico e conectado. É comum dizer apenas que o grafo é uma árvore omitindo o “livre”.
- Floresta: grafo não direcionado acíclico, podendo ou não ser conectado. É um conjunto de árvores.



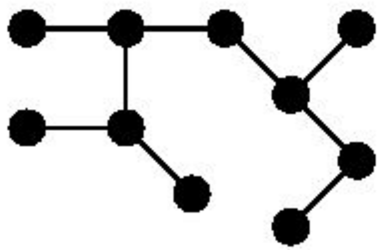
Árvore livre



Floresta

# Árvores e floresta

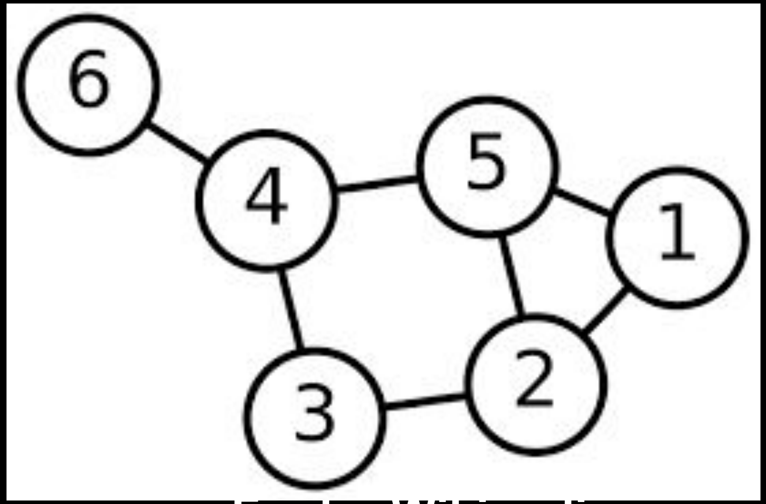
- Quantas arestas têm uma árvore com n vértices?



# Representação de grafos

# Representação de grafos

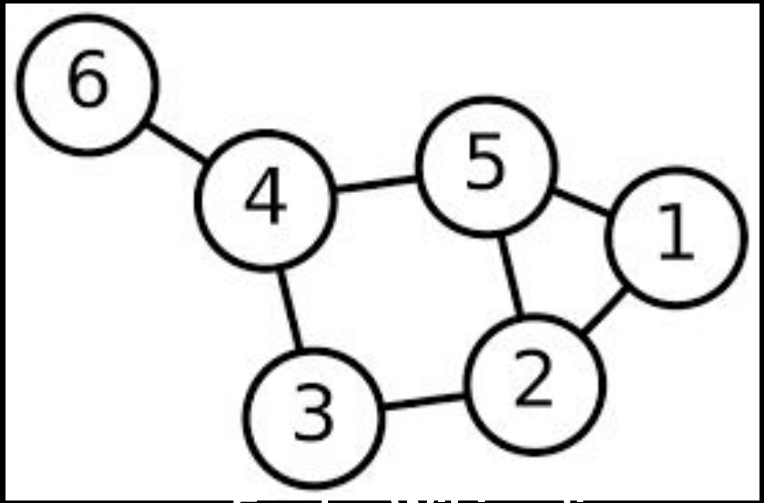
**G=(V,A)**
**V={1,2,3,4,5,6}**
**A={(1,2), (1,5), (2,3), (2,5), (3,4), (4,5), (4,6)}**




**Fonte: Wikipedia**

# Representação de grafos

**G=(V,A)**
**V={1,2,3,4,5,6}**
**A={(1,2), (1,5), (2,3), (2,5), (3,4), (4,5), (4,6)}**




**Fonte: Wikipedia**

**Como um grafo pode ser representado no computador?**

# Representação de grafos

Como uma **matriz de adjacências**
Como uma **coleção de listas de adjacências**

Qual a representação mais eficiente ou mais adequada?

# Representação de grafos

Como uma matriz de adjacências
Como uma coleção de listas de adjacências

Qual a representação mais eficiente ou mais adequada?

Depende

# Matriz de adjacências

**Associar vértices às linhas e colunas da matriz e o elemento da matriz indica se há aresta.**

# Matriz de adjacências

A matriz de adjacência de um grafo G = (V , A) contendo n vértices é uma **matriz n × n**, em que:

– **A[i, j]=1** se e somente se existe um arco do vértice **i** para o vértice **j** .

– **A[i, j]=0** caso contrário.

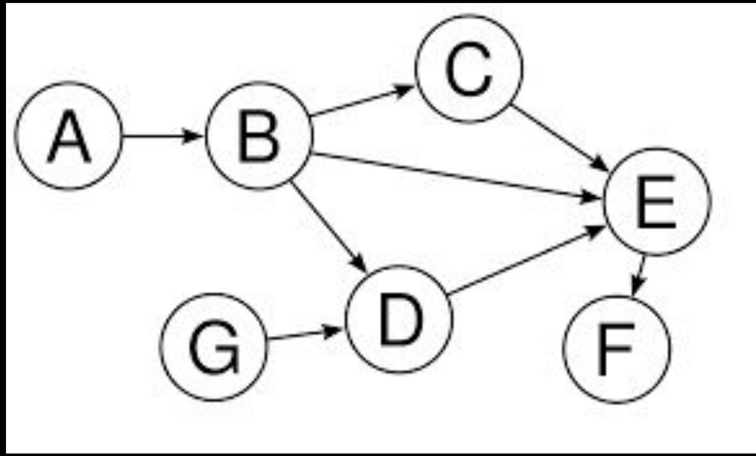
# Matriz de adjacências

Para grafos ponderados:

– A[i, j] contém o rótulo ou peso associado com a aresta do vértice i para o vértice j.

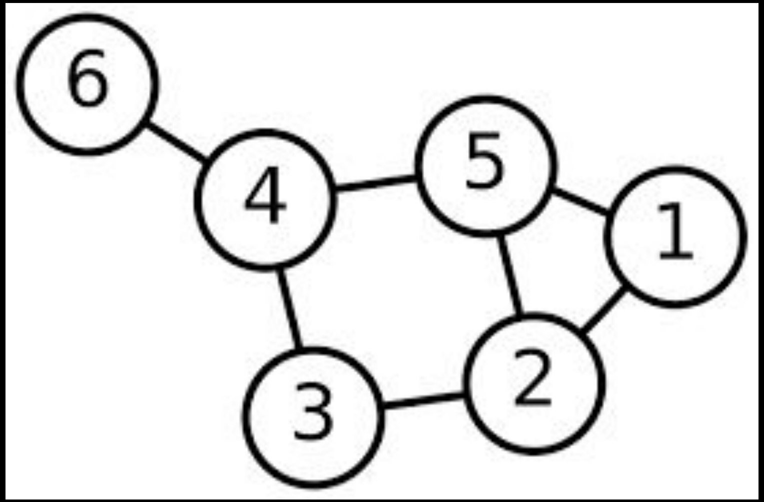
– Se não existir uma aresta de i para j, utilizar um valor que não possa ser usado como rótulo ou peso.

# Matriz de adjacências




| | A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|
| A | | 1 | | | | | |
| B | | | 1 | 1 | 1 | | |
| C | | | | | 1 | | |
| D | | | | | 1 | | |
| E | | | | | | 1 | |
| F | | | | | | | |
| G | | | | 1 | | | |

# Matriz de adjacências




| | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| 1 | | 1 | | | 1 | |
| 2 | 1 | | 1 | | 1 | |
| 3 | | 1 | | 1 | | |
| 4 | | | 1 | | 1 | 1 |
| 5 | 1 | 1 | | 1 | | |
| 6 | | | | 1 | | |

# Matriz de adjacências

Considere grafos grandes (muitos vértices) e esparsos (relativamente poucas arestas)

Matriz estará formada principalmente de zeros!

Grande consumo de memória (desnecessário)!

# Matriz de adjacências

Deve ser utilizada para grafos densos, em que $|A|$ é próximo de $|V|^2$.

O tempo necessário para acessar um elemento é independente de $|V|$ ou $|A|$.

É muito útil para algoritmos em que é necessário saber com rapidez se existe uma aresta ligando dois vértices.

Desvantagem: a matriz necessita $O(|V|^2)$ de espaço.

Como representar grafos grandes e esparsos?

# Coleção de listas de adjacências

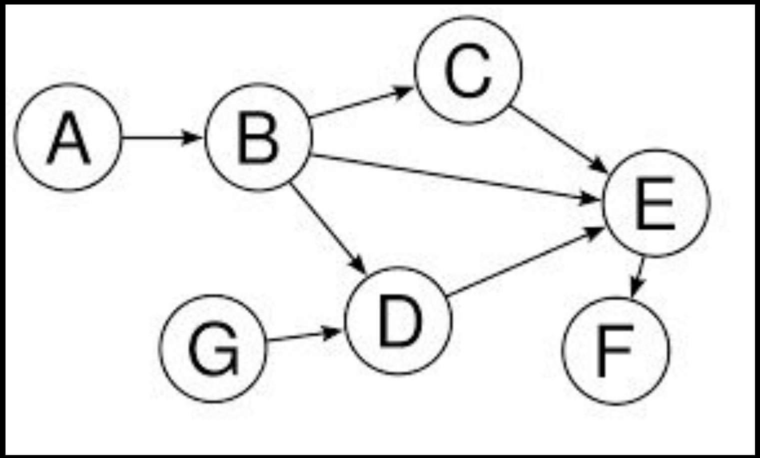
Associar a cada vértice uma lista de vértices adjacentes

# Coleção de listas de adjacências

A representação de um grafo $G = (V,A)$ usando listas de adjacências consiste de:

– Um vetor $Adj$ de $|V|$ listas, uma para cada vértice em $V$.
– Para cada $u \in V$, a lista de adjacências $Adj[u]$ consiste de todos os vértices adjacentes a $u$, i.e., todos os vértices $v$ tais que existe uma aresta $(u,v) \in A$.

# Coleção de listas de adjacências




A → B

B → C → D → E

C → E

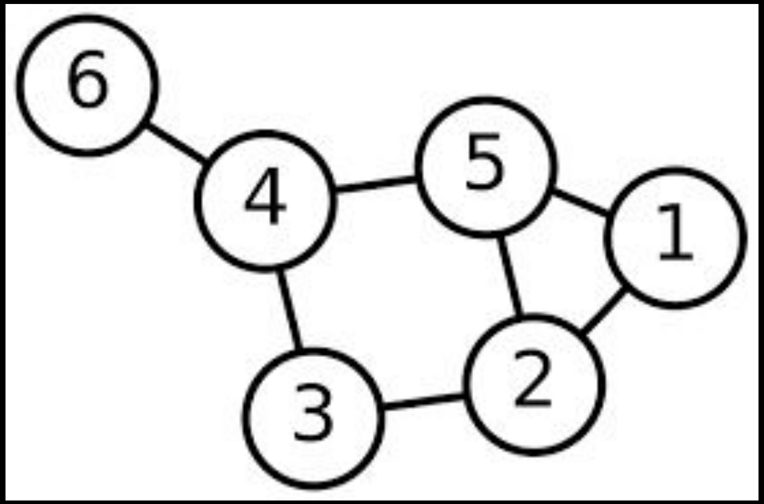
D → E

E → F

F → //

G → D

# Coleção de listas de adjacências




1 → 5 → 2
2 → 3 → 5 → 1
3 → 4 → 2
4 → 3 → 6 → 5
5 → 1 → 2 → 4
6 → 4

# Coleção de listas de adjacências

Os vértices de uma lista de adjacência são em geral armazenados em uma ordem arbitrária.

A soma dos comprimentos de todas as listas de adjacências

– se G é um grafo orientado é: |A|

– se G é um grafo não orientado é: 2*|A|

O espaço requerido por essa representação é:

# Coleção de listas de adjacências

Os vértices de uma lista de adjacência são em geral armazenados em uma ordem arbitrária.

A soma dos comprimentos de todas as listas de adjacências

– se G é um grafo orientado é: |A|

– se G é um grafo não orientado é: 2*|A|

O espaço requerido por essa representação é:

$$O(|V| + |A|)$$

seja o grafo orientado ou não.

# Coleção de listas de adjacências

Representação mais adequada para grafos **esparsos** ( quando |A| é muito menor do que $|V|^2$). É compacta e usualmente utilizada na maioria das aplicações.

**Desvantagem:** pode consumir **O(|V|)** para determinar se existe uma aresta entre o vértice **i** e o vértice **j**, pois podem existir **O(|V|)** vértices na lista de adjacentes do vértice **i.**

# AULA 1

# Conceitos básicos e representação de grafos

Karina Valdivia Delgado