

Projeto e Análise de Algoritmos

Algoritmos e conceitos fundamentais de grafos

Lehilton Pedrosa

Primeiro Semestre de 2023

Conceitos de grafos

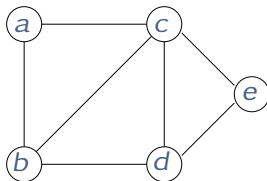
Definição de grafo

Um **grafo** é um par $G = (V, E)$ onde

- ▶ V é um conjunto finito de elementos chamados **vértices** e
- ▶ E é um conjunto finito de pares **não ordenados** de vértices chamados **arestas**.

Exemplo:

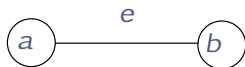
- ▶ $V = \{a, b, c, d, e\}$
- ▶ $E = \{(a, b), (a, c), (b, c), (b, d), (c, d), (c, e), (d, e)\}$



Adjacência e incidência

Considere uma aresta $e = (a, b)$

- ▶ note que para pares não ordenados, temos $(a, b) = (b, a)$
- ▶ desenhamos a aresta como uma linha ligando os vértices
- ▶ dizemos que os vértices a e b são os **extremos** de e
- ▶ e também que a e b são vértices **adjacentes**



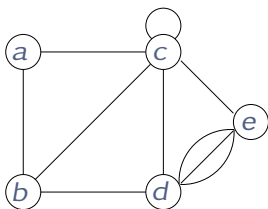
Para enfatizar a relação entre arestas e vértices

- ▶ dizemos que a aresta e é **incidente** nos vértices a e b
- ▶ e que os vértices a e b são incidentes na aresta e

Multigrafo

Um **multigrafo** é uma generalização de grafos que pode conter

- ▶ **laço**: aresta com extremos idênticos
- ▶ **arestas múltiplas**: duas ou mais arestas com o mesmo par de extremos

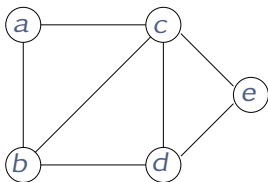


Um grafo é **simples** se ele não tiver laços ou arestas múltiplas

Tamanho do grafo

Considere um grafo $G = (V, E)$

- ▶ denotamos por $|V|$ a cardinalidade do conjunto de vértices
- ▶ e por $|E|$ a cardinalidade do conjunto de arestas
- ▶ no exemplo abaixo temos $|V| = 5$ e $|E| = 7$



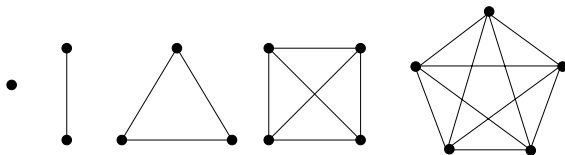
O tamanho do grafo G é dado por $|V| + |E|$

Grafos completos

Um grafo é **completo** se tiver uma aresta (u, v) para todo par de vértices u, v

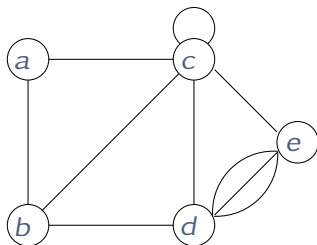
- ▶ se o número de vértices é n , então ele tem $\binom{n}{2}$ arestas
- ▶ portanto, um grafo simples tem **no máximo** $\binom{n}{2}$ arestas

Exemplos de grafos completos:



Grau de um vértice

O **grau** de um vértice v , denotado por $d(v)$ é o número de arestas incidentes a v , com laços contados duas vezes.



$$d(a) = 2$$

$$d(b) = 3$$

$$d(c) = 6$$

$$d(d) = 5$$

$$d(e) = 4$$

Handshaking lemma

Para todo grafo $G = (V, E)$,

$$\sum_{v \in V} d(v) = 2|E|.$$

Alguns nomes

Considere um vértice v de um grafo $G = (V, E)$

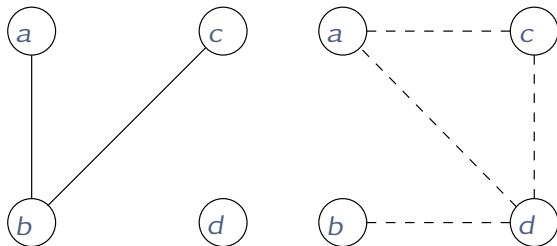
- ▶ se $d(v) = |V| - 1$, dizemos que v é um **vértice universal**
- ▶ se $d(v) = 0$, dizemos que v é um **vértice isolado**

Um grafo completo com três vértices é chamado de **triângulo**

Grafo complementar

O **complemento** de um grafo $G = (V, E)$ é o grafo $\overline{G} = (V, \overline{E})$

- ▶ cujo o conjunto de vértices é o mesmo, V
- ▶ e com $(u, v) \in \overline{E}$ se e somente se $(u, v) \notin E$



Note que $d_{\overline{G}}(v) = |V| - 1 - d_G(v)$.

Consequências da complementaridade

Vamos fazer alguns exercícios

1. Mostre que em uma festa com pelo menos $n \geq 6$ pessoas, existem três pessoas que se conhecem mutuamente ou três pessoas que não se conhecem mutuamente.
2. Suponha que em um grupo S de n pessoas, com $n \geq 4$, vale o seguinte: em qualquer grupo $X \subseteq S$ de 4 pessoas, existe uma que conhece as demais pessoas de X . Mostre que existe uma pessoa em S que conhece todas as demais pessoas de S .

Exercício 1

Considere o grafo G representando as amizades

- ▶ tome um vértice v e lembre-se de que $d_G(v) + d_{\overline{G}}(v) \geq 5$
- ▶ assim ou $d_G(v) \geq 3$, ou $d_{\overline{G}}(v) \geq 3$
- ▶ suponha que em G , o vértice v tem três vizinhos a, b, c
 1. se houver aresta entre vizinhos, ela forma triângulo com v
 2. do contrário, a, b, c formam um triângulo em $\overline{G}$

Exercício 2

Considere o grafo G representando as **não** amizades

- ▶ dois vértices são ligados se as pessoas não se conhecem
- ▶ queremos encontrar um vértice isolado de G
- ▶ deve existir aresta (x, y) , senão não há nada a fazer

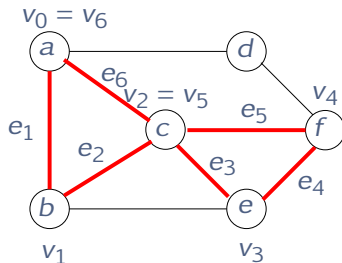
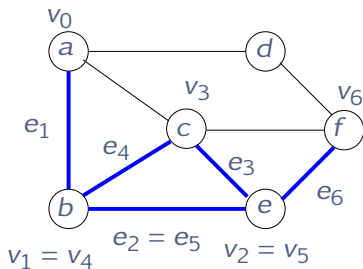
Tome um conjunto $X = \{x, y, z, w\}$

- ▶ ou w conhece $\{x, y, z\}$, ou z conhece $\{x, y, z\}$
- ▶ sem perda de generalidade, w conhece $\{x, y, z\}$
- ▶ assim não há arestas de w a $\{x, y, z\}$
- ▶ mas como z é arbitrário, não há aresta de w a todo z
- ▶ assim, w é vértice isolado

Passeios em grafos

Um **passeio** P de um vértice v_0 a um vértice v_k em um grafo G é uma sequência finita e não vazia $(v_0, e_1, v_1, \dots, e_k, v_k)$

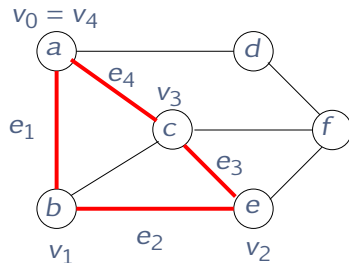
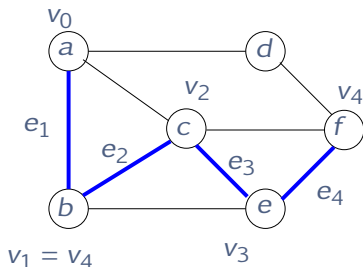
- ▶ cujos elementos são alternadamente vértices e arestas
- ▶ e tal que, para $1 \leq i \leq k$, v_{i-1} e v_i são os extremos de e_i



- ▶ dizemos que v_k é **alcançável** a partir de v_0 através de P
- ▶ se $v_0 = v_k$, então dizemos que P é um **passeio fechado**
- ▶ o **comprimento** de P é o seu número de arestas, ou seja, k

Caminhos e ciclos

- ▶ Um **caminho** é um passeio cujos vértices são distintos.
- ▶ Um **ciclo** é um passeio fechado que possui pelo menos uma aresta e tal que $v_1, \dots, v_k$ são distintos e todas as arestas são distintas.



Refletindo sobre as definições

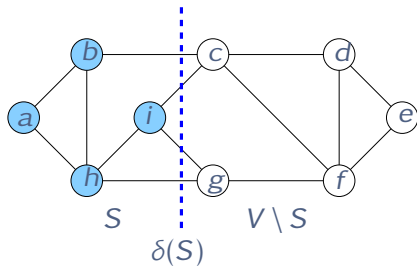
Responda as seguintes questões

1. Seja G um grafo e u, v vértices de G . Mostre que se existe um passeio de u a v em G , então existe um caminho de u a v em G . Por que isto é um resultado interessante?
2. Seja G um grafo e u, v, w vértices de G . Mostre que se em G existem um caminho de u a v e um caminho de v a w então existe um caminho de u a w em G .
3. É verdade que todo passeio fechado contém um ciclo?

Cortes

Seja $G = (V, E)$ um grafo e seja $S \subset V$.

- Denote por $\delta_G(S)$ o conjunto de arestas de G com um extremo em S e outro em $V \setminus S$.



- dizemos que $\delta_G(S)$ é um **corte** de G induzido por S
- se $s \in S$ e $t \in V \setminus S$ dizemos que $\delta_G(S)$ **separa** s de t

Caminhos versus cortes

Lema

Seja G um grafo e sejam s, t vértices distintos de G .

Então exatamente uma das afirmações é verdadeira:

- (a) existe um caminho de s a t em G , ou
- (b) existe um corte $\delta_G(S)$ que separa s de t tal que $\delta_G(S) = \emptyset$.

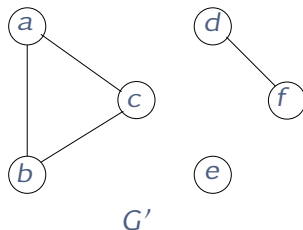
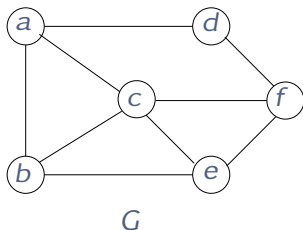
Demonstração:

- ▶ (a) e (b) não podem valer simultaneamente (por quê?)
- ▶ suponha que (a) não vale
- ▶ seja S o conjunto dos vértices alcançáveis por s em G
- ▶ segue que $t \in V \setminus S$ e $\delta_G(S) = \emptyset$

Conexidade

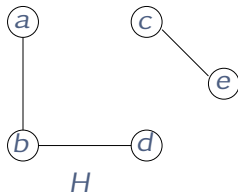
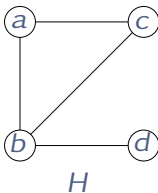
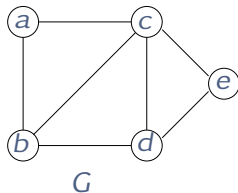
Dizemos que um grafo G é **conexo** se, para qualquer par de vértices u e v de G , existir um caminho de u a v em G .

- ▶ caso contrário, dizemos que é **desconexo**
- ▶ podemos particionar o grafo em **componentes**
- ▶ u e v estão na mesma componente se há caminho de u a v



Subgrafo e subgrafo gerador

- ▶ Um **subgrafo** $H = (V', E')$ de um grafo $G = (V, E)$ é um grafo tal que $V' \subseteq V$ e $E' \subseteq E$.
- ▶ Um **subgrafo gerador** de G é um subgrafo com $V' = V$.



Grafos obtidos a partir de outros grafos

Considere um grafo $G = (V, E)$, uma aresta e e um vértice v

- ▶ $G - e$ é o grafo obtido de G removendo-se e :

$$G - e = (V, E \setminus \{e\})$$

- ▶ $G - v$ é o grafo obtido de G removendo-se v e todas as arestas que incidem em v :

$$G - v = (V \setminus \{v\}, E \setminus \delta(\{v\}))$$

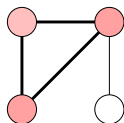
Subgrafo induzido

Considere um grafo $G = (V, E)$ e um subconjunto de vértices S .

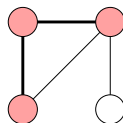
- ▶ O subgrafo de G **induzido** por S , denotado por $G[S]$, é o grafo formado por S e todas as arestas entre vértices S :

$$G[S] = (S, \{(u, v) \in E : u, v \in S\})$$

Exemplos:



INDUZIDO

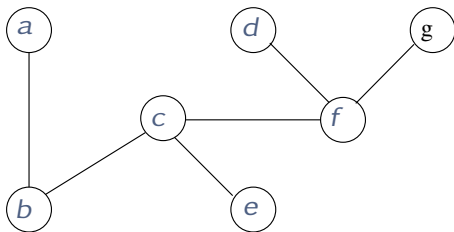


NÃO INDUZIDO

Árvores

Árvores

Um grafo $G = (V, E)$ é uma **árvore** se ele for conexo e não possuir ciclos.



- ▶ um grafo sem ciclos é chamado de **acíclico**
- ▶ uma **folha** de uma árvore G é um vértice de grau 1
- ▶ toda árvore com dois ou mais vértices tem folha (por quê?)

Caracterização de árvores

Teorema

As seguintes afirmações são equivalentes:

- ▶ G é uma árvore.
- ▶ G é conexo e possui exatamente $|V| - 1$ arestas.
- ▶ G é conexo e a remoção de qualquer aresta desconecta o grafo, i.e, ele é conexo minimal.
- ▶ Para todo par de vértices u, v de G , existe um único caminho de u a v em G .

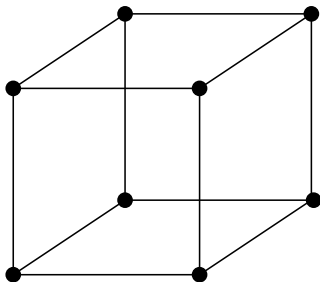
Grafos bipartidos

Uma **bipartição** de um conjunto V é um par (A, B) tal que:

- ▶ $A \cap B = \emptyset$ e
- ▶ $A \cup B = V$.

Um grafo $G = (V, E)$ é **bipartido** se existe uma partição (A, B) de V tal que toda aresta de G tem um extremo em A e outro em B .

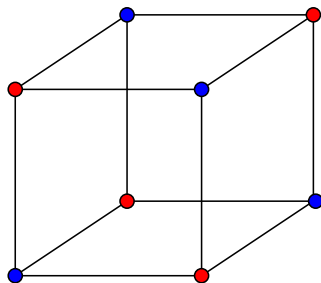
Grafos bipartidos



Esse grafo é bipartido?

- ▶ vamos indicar cada parte com uma cor: **vermelho** ou **azul**

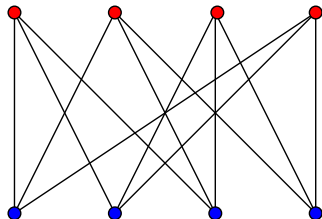
Grafos bipartidos



É bipartido!

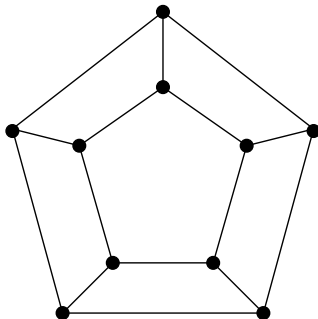
- ▶ um grafo $G = (V, E)$ é bipartido se for possível colorir os vértices de G com **duas cores** de modo que vértices adjacentes tenham cores distintas

Grafos bipartidos



Isto pode ser visto melhor com outro desenho.

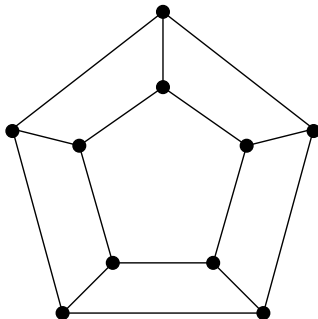
Grafos bipartidos



Este grafo **não** é bipartido.

- ▶ você consegue apresentar uma justificativa simples?

Grafos bipartidos



Condição necessária para um grafo ser bipartido:

- ▶ o grafo não pode conter ciclos de comprimento ímpar
- ▶ é uma condição suficiente? basta não ter ciclos ímpares?
- ▶ vamos mostrar que **sim**!

Teorema

Seja G um grafo. Então G é bipartido se e somente se G não possuir um ciclo ímpar.

Demonstração:

- ▶ já vimos que se G tem um ciclo ímpar, ele não é bipartido
- ▶ assim, resta demonstrar a recíproca
- ▶ podemos supor que G é conexo (por quê?)

Árvore geradora

Fato 1

Todo grafo conexo contém uma árvore geradora.

Demonstração:

- ▶ segue facilmente do próximo resultado
- ▶ demonstre-o como exercício

Lema

Seja G um grafo conexo e seja C um ciclo de G . Se e é uma aresta de C então $G - e$ é conexo.

A recíproca também vale.

Lema

Seja G um grafo conexo e seja e uma aresta de G . Se $G - e$ é conexo então e pertence a algum ciclo de G .

Fato 2

Toda árvore $T = (V, E)$ é um grafo bipartido.

Demonstração:

- ▶ a prova é por indução em $|V|$
- ▶ segue do fato de que toda árvore com pelo menos dois vértices contém uma folha
- ▶ termine a demonstração como exercício

Fato 3

Seja $T = (V, E')$ uma árvore geradora de um grafo $G = (V, E)$. Então para toda aresta $e \in E \setminus E'$ existe um único ciclo em $T + e = (V, E' \cup \{e\})$.

Demonstração:

- ▶ sejam u, v os extremos de e
- ▶ como T é árvore, existe um único caminho P de u a v em T
- ▶ portanto, $P + e$ é o único ciclo em $T + e$

O único ciclo de $T + e$ é chamado de **ciclo fundamental**.

Demonstração do teorema

Agora estamos prontos para demonstrar o teorema.

Teorema

Seja G um grafo. Então G é bipartido se e somente se G não possui um ciclo ímpar.

Demonstração:

- ▶ resta mostrar que se G não tem ciclo ímpar, ele é bipartido
- ▶ suponha que G não contenha um ciclo ímpar
- ▶ construiremos uma bipartição (A, B) de V tal que toda aresta de G tem um extremo em A e outro em B
- ▶ lembrando, podemos supor que G seja conexo

Demonstração do teorema (cont)

- ▶ pelo Fato 1, G contém uma árvore geradora $T = (V, E')$
- ▶ pelo Fato 2, T possui uma bipartição (A, B) de V tal que toda aresta de T tem um extremo em A e outro em B
- ▶ mostraremos que toda aresta de $E \setminus E'$ tem um extremo em A e outro em B
- ▶ isso implica que G é bipartido

Demonstração do teorema (cont)

Seja e uma aresta de $E \setminus E'$

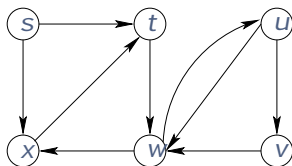
- ▶ pelo Fato 3, existe um único ciclo C em $T + e$ que contém e
- ▶ se os extremos de e são da mesma parte (A ou B),
- ▶ então C é um ciclo ímpar, o que é uma **contradição!**
- ▶ segue que os extremos de e estão em partes distintas
- ▶ que é o que precisávamos

Outros tipos de grafos

Grafo direcionado

Um **grafo direcionado** é definido de forma semelhante, com a diferença que as arestas consistem de **pares ordenados** de vértices.

- ▶ também chamamos essas variantes de **digrafos**
- ▶ muitas vezes chamamos suas arestas de **arcos**



Adjacência de grafos direcionados

Considere uma aresta $e = (u, v)$ de um grafo direcionado G

- ▶ dizemos que e sai de u e entra em v
- ▶ o vértice u é a **cauda** de e
- ▶ e o vértice v é **cabeça** de e

Temos dois tipos de grau para grafos direcionados

- ▶ **grau de saída** $d^+(v)$ é o número de arestas que saem de v
- ▶ **grau de entrada** $d^-(v)$ é o número de arestas que entram em v

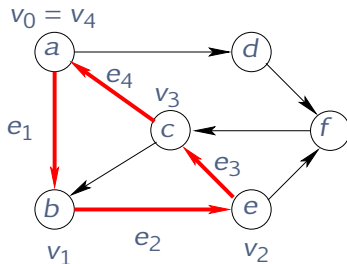
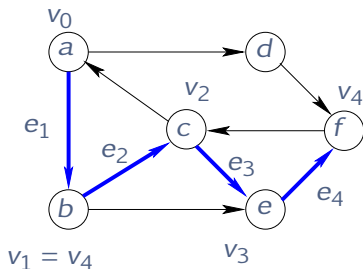
Teorema

Para todo grafo direcionado $G = (V, E)$ temos:

$$\sum_{v \in V} d^+(v) = \sum_{v \in V} d^-(v) = |E|.$$

Passeios em grafos direcionados

Em um **passeio direcionado** de um grafo direcionados todas as arestas seguem o mesmo sentido.



- ▶ definimos **caminhos e ciclos direcionados** analogamente
- ▶ assim com os subgrafos de um grafo direcionado
- ▶ veremos a noção de conexidade para esses grafos depois

Refletindo sobre as definições

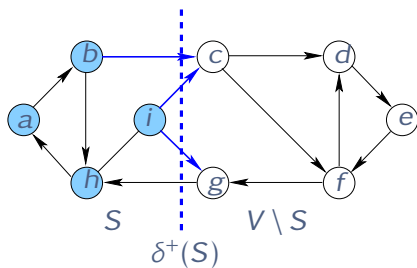
Vamos rever algumas questões, mas para grafos direcionados

1. Seja G um grafo direcionado e u, v vértices de G . Mostre que se existe um passeio de u a v em G , então existe um caminho de u a v em G .
2. Seja G um grafo direcionado e u, v, w vértices de G . Mostre que se em G existem um caminho de u a v e um caminho de v a w então existe um caminho de u a w em G .
3. É verdade que todo passeio fechado em um grafo direcionado contém um ciclo direcionado?

Cortes em grafos direcionados

Seja $G = (V, E)$ um grafo direcionado e seja $S \subset V$.

- Denote por $\delta_G^+(S)$ o conjunto de arestas de G com cauda em S e cabeça em $V \setminus S$.



- dizemos que $\delta_G^+(S)$ é um **corte direcionado**
- se $s \in S$ e $t \in V \setminus S$ dizemos que $\delta_G^+(S)$ **separa** s de t
- note que (g, h) não está no corte $\delta_G^+(S)$

Caminhos versus cortes direcionados

Lema

Seja G um grafo direcionado e sejam s, t vértices distintos de G . Então exatamente uma das afirmações é verdadeira:

- (a) existe um caminho de s a t em G , ou
- (b) existe um corte direcionado $\delta_G^+(S)$ que separa s de t tal que $\delta_G^+(S) = \emptyset$.

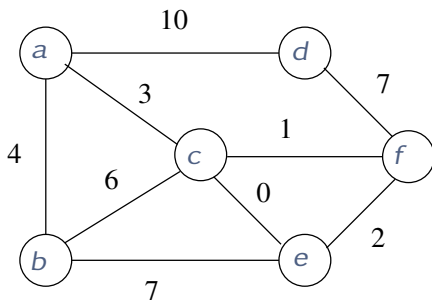
Demonstração:

- ▶ é análoga à do lema para grafos não direcionados
- ▶ faça como exercício

Grafo ponderado

Um grafo é **ponderado** se a cada aresta e do grafo está associado um valor real $w(e)$, denominado peso da aresta.

- ▶ o grafo pode ser direcionado ou não
- ▶ também dizemos que $w(e)$ é o custo da aresta



Representação de grafos

Grafos podem modelar diversas estruturas reais:

1. se computadores são representados por vértices, então as conexões entre eles correspondem a arestas
2. se as cidades forem representadas por vértices, então as estradas correspondem a arestas direcionadas
3. etc.

Queremos construir algoritmos genéricos

- ▶ vamos estudar algoritmos para grafos de modo **abstrato**
- ▶ mas eles são aplicados em problemas **concretos**

Problema do caminho mínimo

- ▶ dadas as cidades, as distâncias entre elas e duas cidades A e B , determinar um **trajeto mais curto** de A até B

Problema da árvore geradora mínima

- ▶ dados os computadores e o custo de conectar cada par de computadores, projetar uma rede interconectando todos os computadores de **menor custo** possível

Problema do emparelhamento máximo

- ▶ dadas vagas de empregos e uma lista de candidatos para cada vaga, determinar uma lista de associações candidato-emprego de **maior tamanho** possível

Problema do caixeiro viajante

- ▶ dadas cidades e as distâncias entre elas, encontrar um rota que visita todas as cidades de **comprimento mínimo**

Problema do carteiro chinês

- ▶ dadas as ruas de um bairro, encontrar uma rota fechada que passa por todas as ruas de **comprimento mínimo**

Representação interna de grafos

Representamos grafos de duas maneiras principais

1. matriz de adjacência
2. listas de adjacência

Qual estrutura de dados escolher?

- ▶ depende do problema sendo tratado
- ▶ e das operações realizadas pelo algoritmo
- ▶ a estrutura escolhida afeta a complexidade do algoritmo

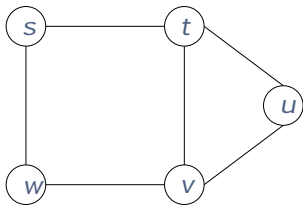
Matriz de adjacência

A **matriz de adjacência** de um grafo simples G é uma matriz quadrada A de ordem $|V|$ tal que

$$A[i,j] = \begin{cases} 1 & \text{se } (i,j) \in E, \\ 0 & \text{caso contrário.} \end{cases}$$

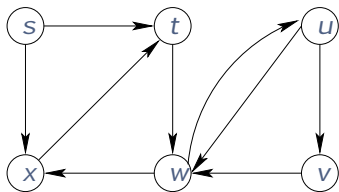
- ▶ o grafo pode ser direcionado ou não
- ▶ se G for não direcionado, então a matriz A é simétrica

Matriz de adjacência



	s	t	w	v	u
s	0	1	1	0	0
t	1	0	0	1	1
w	1	0	0	1	0
v	0	1	1	0	1
u	0	1	0	1	0

Matriz de adjacência



	s	t	u	x	w	v
s	0	1	0	1	0	0
t	0	0	0	0	1	0
u	0	0	0	0	1	1
x	0	1	0	0	0	0
w	0	0	1	1	0	0
v	0	0	0	0	1	0

Listas de adjacência

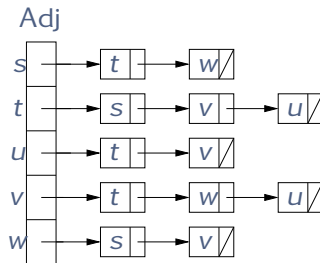
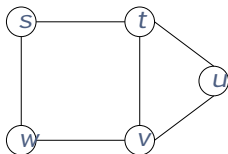
Para representar um grafo $G = (V, E)$ por **listas de adjacências**:

- ▶ criamos uma lista ligada $\text{Adj}[v]$ para cada vértice v
- ▶ adicionamos a $\text{Adj}[v]$ todos os vértices adjacentes a v

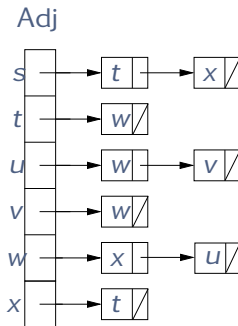
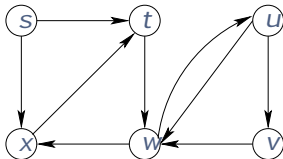
Como representamos uma aresta (u, v) ?

- ▶ se a aresta for direcionado, então v está em $\text{Adj}[u]$
- ▶ se a aresta for **não** direcionada,
 1. então v está em $\text{Adj}[u]$
 2. também u está em $\text{Adj}[v]$

Listas de adjacência



Listas de adjacências



Notação para complexidade

Considere um grafo $G = (V, E)$

- ▶ vamos simplificar a **notação assintótica**
- ▶ escrevemos V e E ao invés de $|V|$ e $|E|$
- ▶ por exemplo, $O(E^2 \log V)$ ao invés de $O(|E|^2 \log |V|)$

Matriz versus listas

A melhor representação depende do algoritmo

1. matriz de adjacência

- ▶ é fácil verificar se (u, v) é uma aresta de G
- ▶ o espaço utilizado é $\Theta(V^2)$
- ▶ adequada para grafos densos (com $|E| = \Theta(V^2)$)

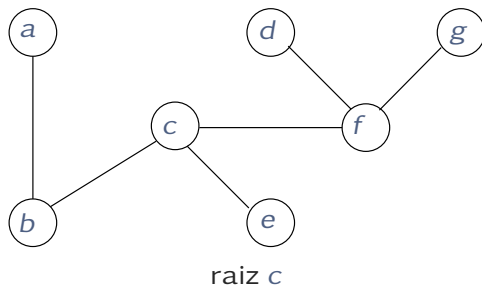
2. listas de adjacência

- ▶ é fácil listar os vértices adjacentes de um dado vértice v
- ▶ o espaço utilizado é $\Theta(V + E)$
- ▶ adequada a grafos esparsos (com $|E| = \Theta(V)$)

- ▶ Há alternativas para representar grafos, mas matrizes e listas de adjacência são as mais usadas.
- ▶ Essas representações podem ser usadas para grafos ponderados, grafos com laços e arestas múltiplas, grafos com pesos nos vértices etc.
- ▶ Para determinados algoritmos é importante manter estruturas de dados adicionais.

Representação de árvores

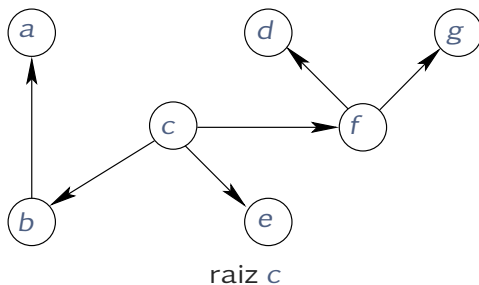
Uma **árvore enraizada** é uma árvore com um vértice especial chamado **raiz**.



Representação de árvores

Uma **árvore direcionada** com raiz r é um grafo direcionado acíclico $T = (V, E)$ tal que:

1. $d^-(r) = 0$,
2. $d^-(v) = 1$ para $v \in V \setminus \{r\}$.

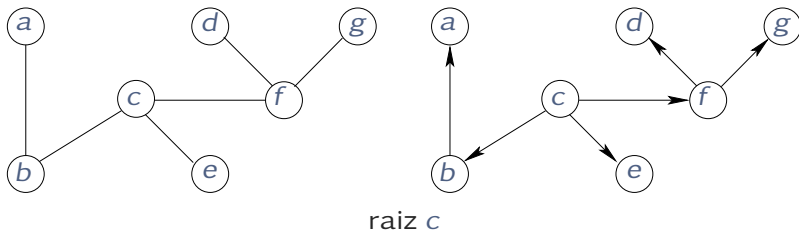


Representação de árvores

Representar uma árvore enraizada com um vetor de predecessores π .

v	a	b	c	d	e	f	g
$\pi[v]$	b	c	N	f	c	c	f

- usamos o símbolo N para indicar a ausência



Alguns detalhes de implementação

- ▶ Nos algoritmos que veremos, usamos a representação de um grafo (direcionado ou não) por listas de adjacências.
- ▶ Em uma implementação real de um algoritmo, provavelmente a representação **não** é dada a priori.
- ▶ Assim, é necessário construir tal representação a partir da dos dados de entrada.
- ▶ Como construir a representação de um grafo depende do formato da entrada.

Exemplo de entrada

Suponha que a entrada é um arquivo texto

- ▶ o arquivo representa um grafo **não** direcionado
- ▶ a primeira linha contém $|V|$ e $|E|$
- ▶ os vértices são numerado de 0 a $|V| - 1$
- ▶ as próximas linhas representam os extremos das arestas

5 7

0 1

0 2

1 2

1 3

2 3

2 4

3 4

Exemplo de construção

CONSTRUIR-ADJ()

- 1 leia n e m
- 2 repita m vezes
- 3 leia a próxima aresta (u, v)
- 4 insira v na lista $\text{Adj}[u]$
- 5 insira u na lista $\text{Adj}[v]$
- 6 devolva Adj

Calculando o quadrado

O **quadrado** de um grafo direcionado $G = (V, E)$ é o grafo direcionado $G^2 = (V, E^2)$ tal que:

- ▶ $(u, v) \in E^2$ se e somente se existe um caminho de comprimento no máximo dois de u a v em G .

Dado G , vamos calcular G^2

1. utilizando matriz de adjacência
2. utilizando listas de adjacência

Quadrado com matriz de adjacência

QUADRADO(M)

```
1   $N \leftarrow M$ 
2  para cada  $u \in V$ 
3      para cada  $v \in V$ 
4          para cada  $w \in V$ 
5              se  $N[u, v] = 0$ 
6                   $N[u, v] \leftarrow M[u, w] \cdot M[w, v]$ 
7  devolva  $N$ 
```

Complexidade: $\Theta(V^3)$

Quadrado com listas de adjacências

QUADRADO(Adj)

- 1 crie uma cópia Adj' de Adj
- 2 para cada $v \in V$
- 3 para cada $u \in Adj[v]$
- 4 concatene uma cópia de $Adj[u]$ a $Adj'[v]$
- 5 para cada $v \in V$
- 6 ordene $Adj'[v]$ usando COUNTING-SORT
- 7 elimine as repetições de $Adj'[v]$ em tempo linear
- 8 devolva Adj'

Complexidade: $\Theta(VE)$