Matemática Discreta

Dirk Hofmann

Departamento de Matemática, Universidade de Aveiro
dirk@ua.pt, http://sweet.ua.pt/dirk/aulas

Gabinete: 11.3.10

Atendimento de dúvidas: Terça, 15:00 – 17:00

Alguns conceitos métricos

Passeios em grafos

Definição

Seja $G = (V, E, \psi)$ um grafo. Um passeio em G é uma sequência

$$P = (v_0, e_1, v_1, e_2, \ldots e_k, v_k) \qquad (k \in \mathbb{N})$$

onde $v_0, v_1, \ldots, v_k \in V$, $e_1, e_2, \ldots, e_k \in E$ e, para cada $i = 1, 2, \ldots, k$, $\psi(e_i) = v_{i-1}v_i$.

Passeios em grafos

Definição

Seja $G = (V, E, \psi)$ um grafo. Um passeio em G é uma sequência

$$P = (v_0, e_1, v_1, e_2, \dots e_k, v_k) \qquad (k \in \mathbb{N})$$

onde $v_0, v_1, \dots, v_k \in V$, $e_1, e_2, \dots, e_k \in E$ e, para cada $i = 1, 2, \dots, k$, $\psi(e_i) = v_{i-1}v_i$.

Neste caso diz-se que P é um passeio entre os vértices v_0 e v_k (ou um passeio-(v_0, v_k)).

Passeios em grafos

Definição

Seja $G = (V, E, \psi)$ um grafo. Um passeio em G é uma sequência

$$P = (v_0, e_1, v_1, e_2, \dots e_k, v_k) \qquad (k \in \mathbb{N})$$

onde $v_0, v_1, \dots, v_k \in V$, $e_1, e_2, \dots, e_k \in E$ e, para cada $i = 1, 2, \dots, k$, $\psi(e_i) = v_{i-1}v_i$.

Neste caso diz-se que P é um passeio entre os vértices v_0 e v_k (ou um passeio-(v_0, v_k)). O vértice v_0 designa-se por vértice inicial do passeio P e v_k designa-se por vértice final do passeio P, os vértices $v_1, \dots, v_{k-1}$ designam-se por vértices intermédios.

Passeios em grafos

Definição

Seja $G = (V, E, \psi)$ um grafo. Um passeio em G é uma sequência

$$P = (v_0, e_1, v_1, e_2, \ldots e_k, v_k) \qquad (k \in \mathbb{N})$$

onde $v_0, v_1, \ldots, v_k \in V$, $e_1, e_2, \ldots, e_k \in E$ e, para cada $i = 1, 2, \ldots, k$, $\psi(e_i) = v_{i-1}v_i$.

Neste caso diz-se que P é um passeio entre os vértices v_0 e v_k (ou um passeio-(v_0, v_k)). O vértice v_0 designa-se por vértice inicial do passeio P e v_k designa-se por vértice final do passeio P, os vértices $v_1, \ldots, v_{k-1}$ designam-se por vértices intermédios.

Nota

Num grafo simples, um passeio é determinado pela sequência dos sucessivos vértices, isto é, basta considerar

$$P = (v_0, v_1, \ldots, v_k).$$

Trajetos, caminhos, circuitos e ciclos

Definição

Seja $G = (V, E, \psi)$ um grafo.

Trajetos, caminhos, circuitos e ciclos

Definição

Seja $G = (V, E, \psi)$ um grafo.

- Um trajeto é um passeio sem arestas repetidas.

Trajetos, caminhos, circuitos e ciclos

Definição

Seja $G = (V, E, \psi)$ um grafo.

- Um trajeto é um passeio sem arestas repetidas.
- Um trajeto diz-se fechado quando tem pelo menos uma aresta e o vértice inicial coincide com o vértice final $(v_0 = v_k)$. Um trajeto fechado diz-se também circuito.

Trajetos, caminhos, circuitos e ciclos

Definição

Seja $G = (V, E, \psi)$ um grafo.

- Um trajeto é um passeio sem arestas repetidas.
- Um trajeto diz-se fechado quando tem pelo menos uma aresta e o vértice inicial coincide com o vértice final $(v_0 = v_k)$. Um trajeto fechado diz-se também circuito.
- Um caminho é um trajeto que não repete vértices.

Trajetos, caminhos, circuitos e ciclos

Definição

Seja $G = (V, E, \psi)$ um grafo.

- Um trajeto é um passeio sem arestas repetidas.
- Um trajeto diz-se fechado quando tem pelo menos uma aresta e o vértice inicial coincide com o vértice final ($v_0 = v_k$). Um trajeto fechado diz-se também circuito.
- Um caminho é um trajeto que não repete vértices.
- Se $P = (v_0, e_1, v_1, e_2, \ldots e_k, v_k)$ é um caminho com $k \geq 2$ e e_{k+1} é uma aresta com pontos extremos v_k e v_0, então

  $$C = (v_0, e_1, v_1, e_2, \ldots e_k, v_k, e_{k+1}, v_0)$$

  diz-se ciclo.

Comprimento de passeios

Definição

Seja $G = (V, E, \psi)$ um grafo e seja $P = (v_0, e_1, v_1, e_2, \ldots e_k, v_k)$ um passeio de G. Então, o comprimento de P é

$$\mathrm{comp}(P) = k;$$

ou seja, $\mathrm{comp}(P)$ é o número de arestas (com eventual repetição) que o constitui.

Comprimento de passeios

Definição

Seja $G = (V, E, \psi)$ um grafo e seja $P = (v_0, e_1, v_1, e_2, \ldots e_k, v_k)$ um passeio de G. Então, o comprimento de P é

$$\text{comp}(P) = k;$$

ou seja, $\text{comp}(P)$ é o número de arestas (com eventual repetição) que o constitui.

Nota

No caso dos caminhos e dos trajetos, o comprimento coincide com o número de arestas.

Comprimento de passeios

Definição

Seja $G = (V, E, \psi)$ um grafo e seja $P = (v_0, e_1, v_1, e_2, \ldots e_k, v_k)$ um passeio de G. Então, o comprimento de P é

$$\text{comp}(P) = k;$$

ou seja, $\text{comp}(P)$ é o número de arestas (com eventual repetição) que o constitui.

Nota

No caso dos caminhos e dos trajetos, o comprimento coincide com o número de arestas.

Exemplos

Uma aresta é um caminho de comprimento 1 e um vértice é um caminho de comprimento 0.

Distância entre vértices

Definição

Seja $G = (V, E, \psi)$ um grafo (finito). Para $x, y \in V$, consideramos o conjunto $\mathcal{P}_{x,y} = \{\text{todos os caminhos entre } x \text{ e } y\}$.

Designa-se por distância entre vértices de G a função

$$\begin{aligned} \operatorname{dist}\colon V \times V &\longrightarrow \{0, 1, \dots, \nu(G), \infty\} \\ (x, y) &\longmapsto \begin{cases} \min\{\operatorname{comp}(P) \mid P \in \mathcal{P}_{x,y}\} & \text{se } \mathcal{P}_{x,y} \neq \varnothing, \\ \infty & \text{se } \mathcal{P}_{x,y} = \varnothing. \end{cases} \end{aligned}$$

Distância entre vértices

Definição

Seja $G = (V, E, \psi)$ um grafo (finito). Para $x, y \in V$, consideramos o conjunto $\mathcal{P}_{x,y} = \{$todos os caminhos entre x e $y\}$.

Designa-se por distância entre vértices de G a função

$$\begin{aligned} \mathrm{dist}\colon V \times V &\longrightarrow \{0, 1, \ldots, \nu(G), \infty\} \\ (x, y) &\longmapsto \begin{cases} \min\{\mathrm{comp}(P) \mid P \in \mathcal{P}_{x,y}\} & \text{se } \mathcal{P}_{x,y} \neq \varnothing, \\ \infty & \text{se } \mathcal{P}_{x,y} = \varnothing. \end{cases} \end{aligned}$$

Nota

Tem-se

$$\mathrm{dist}(x, x) = 0, \quad \mathrm{dist}(x, y) + \mathrm{dist}(y, z) \geq \mathrm{dist}(x, z),$$

e $\mathrm{dist}(x, y) = \mathrm{dist}(y, x)$, para todos os $x, y, z \in V$.

Existem caminhos longos. . .

Teorema

Seja $G = (V, E)$ um grafo simples finito.

Existem caminhos longos...

Teorema

Seja $G = (V, E)$ um grafo simples finito.

- *G contém um caminho P tal que $comp(P) \geq \delta(G)$.*

Existem caminhos longos...

Teorema

Seja $G = (V, E)$ *um grafo simples finito.*

- G *contém um caminho* P *tal que* $comp(P) \geq \delta(G)$.
- *Se* $\delta(G) \geq 2$, *então* G *contém um ciclo* C *tal que* $comp(C) \geq \delta(G) + 1$.

Existem caminhos longos...

Teorema

Seja $G = (V, E)$ um grafo simples finito.

- *G contém um caminho P tal que $comp(P) \geq \delta(G)$.*
- *Se $\delta(G) \geq 2$, então G contém um ciclo C tal que $comp(C) \geq \delta(G) + 1$.*

Demonstração.

Seja $P = (v_0, v_1, \ldots, v_k)$ um caminho de maior comprimento em G.

$\square$

Existem caminhos longos. . .

Teorema

Seja $G = (V, E)$ um grafo simples finito.

- *G contém um caminho P tal que $comp(P) \geq \delta(G)$.*
- *Se $\delta(G) \geq 2$, então G contém um ciclo C tal que $comp(C) \geq \delta(G) + 1$.*

Demonstração.

Seja $P = (v_0, v_1, \ldots, v_k)$ um caminho de maior comprimento em G. Portanto, todos os vizinhos de v_k pertencem ao caminho (senão, podia-se prolongar o caminho),

$\square$

Existem caminhos longos...

Teorema

Seja $G = (V, E)$ um grafo simples finito.

- *G contém um caminho P tal que $comp(P) \geq \delta(G)$.*
- *Se $\delta(G) \geq 2$, então G contém um ciclo C tal que $comp(C) \geq \delta(G) + 1$.*

Demonstração.

Seja $P = (v_0, v_1, \ldots, v_k)$ um caminho de maior comprimento em G. Portanto, todos os vizinhos de v_k pertencem ao caminho (senão, podia-se prolongar o caminho), portanto,

$$\mathrm{comp}(P) \geq d(v_k) \geq \delta(G).$$

$\square$

Existem caminhos longos. . .

Teorema

Seja $G = (V, E)$ um grafo simples finito.

- *G contém um caminho P tal que $comp(P) \geq \delta(G)$.*
- *Se $\delta(G) \geq 2$, então G contém um ciclo C tal que $comp(C) \geq \delta(G) + 1$.*

Demonstração.

Seja $P = (v_0, v_1, \ldots, v_k)$ um caminho de maior comprimento em G. Portanto, todos os vizinhos de v_k pertencem ao caminho (senão, podia-se prolongar o caminho), portanto,

$$\text{comp}(P) \geq d(v_k) \geq \delta(G).$$

Seja $i_0 = \min\{i \mid v_i v_k \in E\}$.

$\square$

Existem caminhos longos. . .

Teorema

Seja $G = (V, E)$ um grafo simples finito.

- *G contém um caminho P tal que $comp(P) \geq \delta(G)$.*
- *Se $\delta(G) \geq 2$, então G contém um ciclo C tal que $comp(C) \geq \delta(G) + 1$.*

Demonstração.

Seja $P = (v_0, v_1, \ldots, v_k)$ um caminho de maior comprimento em G. Portanto, todos os vizinhos de v_k pertencem ao caminho (senão, podia-se prolongar o caminho), portanto,

$$\text{comp}(P) \geq d(v_k) \geq \delta(G).$$

Seja $i_0 = \min\{i \mid v_i v_k \in E\}$. Então, $C = (v_i, v_{i+1}, \ldots, v_k, v_i)$ é um ciclo de comprimento $d(v_k) + 1 \geq \delta(G) + 1$. $\square$

Cintura e excentricidade

Definição

Seja $G = (V, E, \psi)$ um grafo finito.

Cintura e excentricidade

Definição

Seja $G = (V, E, \psi)$ um grafo finito.

- A cintura $g(G)$ de G é o comprimento do circuito de menor comprimento em G se existe pelo menos um circuito em G; caso contrario $g(G) = \infty$.

Cintura e excentricidade

Definição

Seja $G = (V, E, \psi)$ um grafo finito.

- A cintura $g(G)$ de G é o comprimento do circuito de menor comprimento em G se existe pelo menos um circuito em G; caso contrario $g(G) = \infty$.
- Seja $v \in V$. A maior distância entre v e todos os vértices de G designa-se por excentricidade de v e denota-se por $e(v)$. Mais formalmente (se $V \neq \varnothing$),

$$e(v) = \max_{u \in V} dist_G(u, v).$$

Cintura e excentricidade

Definição

Seja $G = (V, E, \psi)$ um grafo finito.

- A cintura $g(G)$ de G é o comprimento do circuito de menor comprimento em G se existe pelo menos um circuito em G; caso contrario $g(G) = \infty$.
- Seja $v \in V$. A maior distância entre v e todos os vértices de G designa-se por excentricidade de v e denota-se por $e(v)$. Mais formalmente (se $V \neq \varnothing$),

$$e(v) = \max_{u \in V} dist_G(u, v).$$

- A maior excentricidade dos seus vértices designa-se por diâmetro de G e denota-se por $\mathrm{diam}(G)$.

Cintura e excentricidade

Definição

Seja $G = (V, E, \psi)$ um grafo finito.

- A cintura $g(G)$ de G é o comprimento do circuito de menor comprimento em G se existe pelo menos um circuito em G; caso contrario $g(G) = \infty$.
- Seja $v \in V$. A maior distância entre v e todos os vértices de G designa-se por excentricidade de v e denota-se por $e(v)$. Mais formalmente (se $V \neq \emptyset$),

$$e(v) = \max_{u \in V} dist_G(u, v).$$

- A maior excentricidade dos seus vértices designa-se por diâmetro de G e denota-se por $\mathrm{diam}(G)$.
- A menor excentricidade dos vértices de G designa-se por raio e denota-se por $r(G)$.

Um exemplo

Exemplo

Seja $G = (V, E, \psi)$ *um grafo finito com* $V \neq \varnothing$*. Então,*

$$r(G) \leq \operatorname{diam}(G) \leq 2r(G).$$

Um exemplo

Exemplo

Seja $G = (V, E, \psi)$ *um grafo finito com* $V \neq \varnothing$*. Então,*

$$r(G) \leq \mathrm{diam}(G) \leq 2r(G).$$

Recordamos que:

Um exemplo

Exemplo

Seja $G = (V, E, \psi)$ *um grafo finito com* $V \neq \varnothing$*. Então,*

$$r(G) \leq \operatorname{diam}(G) \leq 2r(G).$$

Recordamos que:

- $r(G) = \min_{x \in V} \max_{y \in V} d(x, y)$.

Um exemplo

Exemplo

Seja $G = (V, E, \psi)$ *um grafo finito com* $V \neq \varnothing$*. Então,*

$$r(G) \leq \mathrm{diam}(G) \leq 2r(G).$$

Recordamos que:

- $r(G) = \min_{x \in V} \max_{y \in V} d(x, y)$.
- $\mathrm{diam}(G) = \max_{x,y \in X} d(x, y)$.

Um exemplo

Exemplo

Seja $G = (V, E, \psi)$ *um grafo finito com* $V \neq \varnothing$*. Então,*

$$r(G) \leq \operatorname{diam}(G) \leq 2r(G).$$

Recordamos que:

- $r(G) = \min_{x\in V} \max_{y\in V} d(x, y)$.
- $\operatorname{diam}(G) = \max_{x,y\in X} d(x, y)$.
- $d(x, y) =$ comprimento do menor caminho (ou ∞).

Um exemplo

Exemplo

Seja $G = (V, E, \psi)$ *um grafo finito com* $V \neq \varnothing$. *Então,*

$$r(G) \leq \operatorname{diam}(G) \leq 2r(G).$$

Recordamos que:

- $r(G) = \min_{x \in V} \max_{y \in V} d(x, y)$.
- $\operatorname{diam}(G) = \max_{x,y \in X} d(x, y)$.
- $d(x, y) =$ comprimento do menor caminho (ou ∞).

Logo, $r(G) \leq \operatorname{diam}(G)$.

Um exemplo

Exemplo

Seja $G = (V, E, \psi)$ *um grafo finito com* $V \neq \varnothing$*. Então,*

$$r(G) \leq \operatorname{diam}(G) \leq 2r(G).$$

Recordamos que:

- $r(G) = \min_{x \in V} \max_{y \in V} d(x, y)$.
- $\operatorname{diam}(G) = \max_{x,y \in X} d(x, y)$.
- $d(x, y) =$ comprimento do menor caminho (ou ∞).

Logo, $r(G) \leq \operatorname{diam}(G)$.

Caso 1: Suponhamos que existem $x, y \in V$ com $d(x, y) = \infty$. Então, para todo o $z \in V$, $d(z, x) = \infty$ ou $d(z, y) = \infty$ e por isso $r(G) = \infty$ e $\operatorname{diam}(G) = \infty$.

Um exemplo

Exemplo

Seja $G = (V, E, \psi)$ um grafo finito com $V \neq \varnothing$. Então,

$$r(G) \leq \operatorname{diam}(G) \leq 2r(G).$$

Recordamos que:

- $r(G) = \min_{x \in V} \max_{y \in V} d(x, y)$.
- $\operatorname{diam}(G) = \max_{x,y \in X} d(x, y)$.
- $d(x, y) =$ comprimento do menor caminho (ou ∞).

Logo, $r(G) \leq \operatorname{diam}(G)$.

Caso 2: Suponhamos que $d(x, y) < \infty$, para todos os $x, y \in V$.

Um exemplo

Exemplo

Seja $G = (V, E, \psi)$ *um grafo finito com* $V \neq \varnothing$*. Então,*

$$r(G) \leq \mathrm{diam}(G) \leq 2r(G).$$

Recordamos que:

- $r(G) = \min_{x \in V} \max_{y \in V} d(x, y)$.
- $\mathrm{diam}(G) = \max_{x,y \in X} d(x, y)$.
- $d(x, y) =$ comprimento do menor caminho (ou ∞).

Logo, $r(G) \leq \mathrm{diam}(G)$.

Caso 2: Suponhamos que $d(x, y) < \infty$, para todos os $x, y \in V$. Sejam x, y os vértices com a maior distância $d(x, y) = \mathrm{diam}(G)$ e seja z o vértice com a menor excentricidade $e(z) = r(G)$.

Um exemplo

Exemplo

Seja $G = (V, E, \psi)$ *um grafo finito com* $V \neq \varnothing$. *Então,*

$$r(G) \leq \mathrm{diam}(G) \leq 2r(G).$$

Recordamos que:

- $r(G) = \min_{x \in V} \max_{y \in V} d(x, y)$.
- $\mathrm{diam}(G) = \max_{x,y \in X} d(x, y)$.
- $d(x, y) =$ comprimento do menor caminho (ou ∞).

Logo, $r(G) \leq \mathrm{diam}(G)$.

Caso 2: Suponhamos que $d(x, y) < \infty$, para todos os $x, y \in V$. Sejam x, y os vértices com a maior distância $d(x, y) = \mathrm{diam}(G)$ e seja z o vértice com a menor excentricidade $e(z) = r(G)$. Portanto:

$$\mathrm{diam}(G) = d(x, y) \leq d(x, z) + (z, y) \leq 2e(z) = 2r(G).$$

Conexidade

Relação de conexidade

Definição

Seja $G = (V, E, \psi)$ um grafo. Os vértices $u, v \in V$ dizem-se conexos se existe um caminho entre eles em G.

Relação de conexidade

Definição

Seja $G = (V, E, \psi)$ um grafo. Os vértices $u, v \in V$ dizem-se conexos se existe um caminho entre eles em G. O grafo G com pelo menos um vértice diz-se conexo quando todos os seus vértices são conexos.

Relação de conexidade

Definição

Seja $G = (V, E, \psi)$ um grafo. Os vértices $u, v \in V$ dizem-se conexos se existe um caminho entre eles em G. O grafo G com pelo menos um vértice diz-se conexo quando todos os seus vértices são conexos. Um grafo não conexo diz-se desconexo.

Relação de conexidade

Definição

Seja $G = (V, E, \psi)$ um grafo. Os vértices $u, v \in V$ dizem-se conexos se existe um caminho entre eles em G. O grafo G com pelo menos um vértice diz-se conexo quando todos os seus vértices são conexos. Um grafo não conexo diz-se desconexo.

Nota

A *relação de conexidade* definida por

$$x \sim y \text{ quando } x \text{ e } y \text{ são conexos}$$

é uma relação de equivalência em V.

Relação de conexidade

Definição

Seja $G = (V, E, \psi)$ um grafo. Os vértices $u, v \in V$ dizem-se conexos se existe um caminho entre eles em G. O grafo G com pelo menos um vértice diz-se conexo quando todos os seus vértices são conexos. Um grafo não conexo diz-se desconexo.

Nota

A *relação de conexidade* definida por

$$x \sim y \text{ quando } x \text{ e } y \text{ são conexos}$$

é uma relação de equivalência em V.

Definição

Os subgrafos induzidos pelas classes de equivalência da relação de conexidade dizem-se componentes conexas.

Relação de conexidade

Definição

Seja $G = (V, E, \psi)$ um grafo. Os vértices $u, v \in V$ dizem-se conexos se existe um caminho entre eles em G. O grafo G com pelo menos um vértice diz-se conexo quando todos os seus vértices são conexos. Um grafo não conexo diz-se desconexo.

Nota

A *relação de conexidade* definida por

$$x \sim y \text{ quando } x \text{ e } y \text{ são conexos}$$

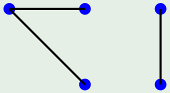
é uma relação de equivalência em V.

Definição

Os subgrafos induzidos pelas classes de equivalência da relação de conexidade dizem-se componentes conexas. O número de componentes conexas de G denota-se por $\mathrm{cc}(G)$.

Relação de conexidade

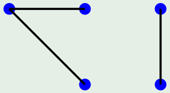
Exemplo



$$cc(G) = 2.$$

Relação de conexidade

Exemplo

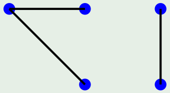


$\text{cc}(G) = 2.$

Nota

Relação de conexidade

Exemplo



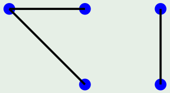
$$\mathrm{cc}(G) = 2.$$

Nota

- As componentes conexas são precisamente os subgrafos (induzidos) conexos *maximais*.

Relação de conexidade

Exemplo



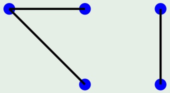
$\mathrm{cc}(G) = 2.$

Nota

- As componentes conexas são precisamente os subgrafos (induzidos) conexos *maximais*.
- Um grafo G é conexo se e só se $\mathrm{cc}(G) = 1$.

Relação de conexidade

Exemplo



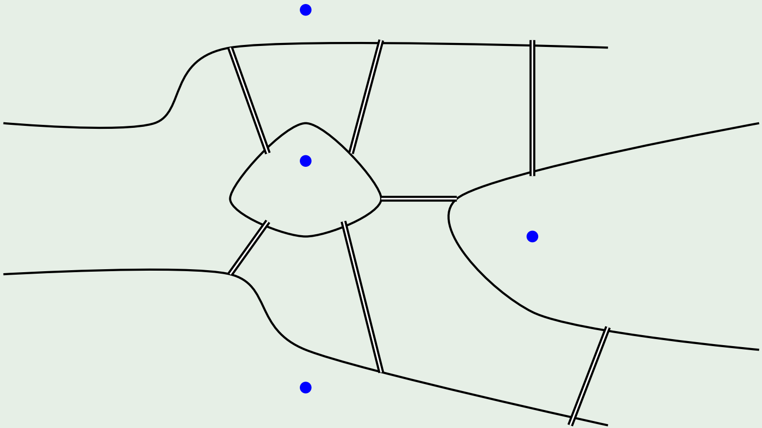
$\mathrm{cc}(G) = 2.$

Nota

- As componentes conexas são precisamente os subgrafos (induzidos) conexos *maximais*.
- Um grafo G é conexo se e só se $\mathrm{cc}(G) = 1$.
- Seja $G = (V, E, \psi)$ um grafo conexo de ordem n. Então, $|E| \geq n - 1$ (ver a solução do exercício 25).

Voltando ao Königsberg

Exemplo



Voltando ao Königsberg

Definição

Seja $G = (V, E, \psi)$ um grafo finito. Um circuito em G diz-se circuito de Euler quando contém todas as arestas de G.

Voltando ao Königsberg

Definição

Seja $G = (V, E, \psi)$ um grafo finito. Um circuito em G diz-se circuito de Euler quando contém todas as arestas de G.

Teorema

Seja G um grafo finito e conexo. Então, G tem um circuito de Euler se e só se todos os vértices de G tem grau par.

Voltando ao Königsberg

Teorema

Seja G um grafo finito e conexo. Então, G tem um circuito de Euler se e só se todos os vértices de G tem grau par.

Demonstração.

$\square$

Voltando ao Königsberg

Teorema

Seja G um grafo finito e conexo. Então, G tem um circuito de Euler se e só se todos os vértices de G tem grau par.

Demonstração.

Suponha que G tem um circuito de Euler, digamos

$$P: \quad v_0 \overset{e_1}{\text{———}} v_1 \cdots\cdots \bullet \cdots\cdots v_{k-1} \overset{e_k}{\text{———}} v_0$$

□

Voltando ao Königsberg

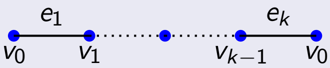
Teorema

Seja G um grafo finito e conexo. Então, G tem um circuito de Euler se e só se todos os vértices de G tem grau par.

Demonstração.

Suponha que G tem um circuito de Euler, digamos

P:



Se um vértice v aparece n vezes em P, então $d(v) = 2n$ é par.

$\square$

Voltando ao Königsberg

Teorema

Seja G um grafo finito e conexo. Então, G tem um circuito de Euler se e só se todos os vértices de G tem grau par.

Demonstração.

Suponha agora que todos os vértices de G tem grau par.

$\square$

Teorema

Seja G um grafo finito e conexo. Então, G tem um circuito de Euler se e só se todos os vértices de G tem grau par.

Demonstração.

Suponha agora que todos os vértices de G tem grau par. Seja

$$P:\quad v_0 \overset{e_1}{\text{———}} v_1 \cdots\cdots \bullet \cdots\cdots v_{k-1} \overset{e_k}{\text{———}} v_k$$

um trajeto de maior comprimento em G.

$\square$

Voltando ao Königsberg

Teorema

Seja G um grafo finito e conexo. Então, G tem um circuito de Euler se e só se todos os vértices de G tem grau par.

Demonstração.

Suponha agora que todos os vértices de G tem grau par. Seja

$$P: \quad v_0 \overset{e_1}{\text{———}} v_1 \cdots\cdots \bullet \cdots\cdots v_{k-1} \overset{e_k}{\text{———}} v_k$$

um trajeto de maior comprimento em G. Logo, P contém todas as arestas com um vértice em v_k. Logo, como $d(v_k)$ é par, $v_0 = v_k$.

$\square$

Voltando ao Königsberg

Teorema

Seja G um grafo finito e conexo. Então, G tem um circuito de Euler se e só se todos os vértices de G tem grau par.

Demonstração.

Suponha agora que todos os vértices de G tem grau par. Seja

$$P:\quad v_0 \overset{e_1}{\frac{\qquad}{}} v_1 \cdots\cdots \bullet \cdots\cdots v_{k-1} \overset{e_k}{\frac{\qquad}{}} v_k$$

um trajeto de maior comprimento em G. Logo, P contém todas as arestas com um vértice em v_k. Logo, como $d(v_k)$ é par, $v_0 = v_k$. Suponha que existe uma aresta fora de P; neste caso existe uma aresta $v \frac{\qquad}{} v_i$ fora de P com v_i em P.

□

Voltando ao Königsberg

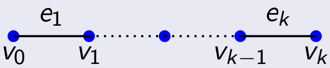
Teorema

Seja G um grafo finito e conexo. Então, G tem um circuito de Euler se e só se todos os vértices de G tem grau par.

Demonstração.

Suponha agora que todos os vértices de G tem grau par. Seja

P:



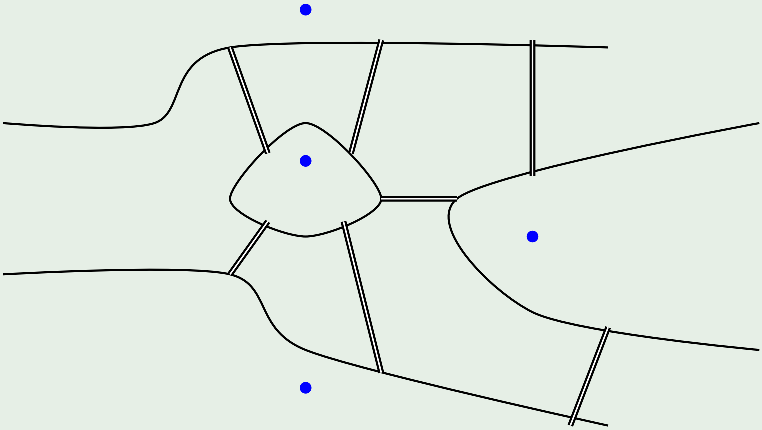
um trajeto de maior comprimento em G. Logo, P contém todas as arestas com um vértice em v_k. Logo, como $d(v_k)$ é par, $v_0 = v_k$. Suponha que existe uma aresta fora de P; neste caso existe uma aresta $v \bullet\!\!-\!\!-\!\!\bullet\, v_i$ fora de P com v_i em P. Então,

P:

é um trajeto mais comprido, uma contradição. $\square$

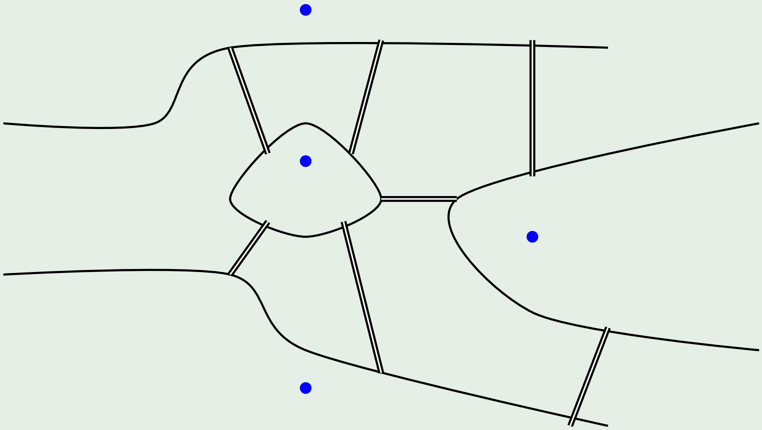
Voltando ao Königsberg

Exemplo



Voltando ao Königsberg

Exemplo



Os vértices tem grau 3, 5, 3 e 3, respetivamente.

Voltando ao Königsberg

Definição

Seja $G = (V, E, \psi)$ um grafo finito. Um trajeto em G diz-se trajeto de Euler quando contém todas as arestas de G.

Voltando ao Königsberg

Definição

Seja $G = (V, E, \psi)$ um grafo finito. Um trajeto em G diz-se trajeto de Euler quando contém todas as arestas de G.

Teorema

Seja G um grafo finito e conexo. Então, G tem um trajeto de Euler se e só o número de vértices de grau ímpar é 0 ou 2.

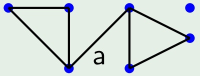
Pontes

Definição

Seja $G = (V, E, \psi)$ um grafo. Uma aresta $a \in E$ diz-se uma ponte (ou uma aresta de corte) quando $\mathrm{cc}(G - a) > \mathrm{cc}(G)$.

Exemplo

G:



A arresta a é uma ponte de G.

Pontes

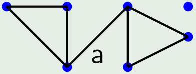
Definição

Seja $G = (V, E, \psi)$ um grafo. Uma aresta $a \in E$ diz-se uma ponte (ou uma aresta de corte) quando $\mathrm{cc}(G - a) > \mathrm{cc}(G)$.

Ou seja, a é uma ponte de G se a eliminação de a aumenta o número de componentes de G.

Exemplo

G:



A arresta a é uma ponte de G.

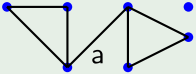
Pontes

Definição

Seja $G = (V, E, \psi)$ um grafo. Uma aresta $a \in E$ diz-se uma ponte (ou uma aresta de corte) quando $\mathrm{cc}(G - a) > \mathrm{cc}(G)$.

Exemplo

G:



A arresta a é uma ponte de G.

Teorema

Sejam $G = (V, E, \psi)$ um grafo e $a \in E$ com $\psi(a) = \{u, v\}$. Então, as seguintes afirmações são equivalentes:

(i) *A aresta a é uma ponte de G*

(ii) $\mathrm{cc}(G - a) = \mathrm{cc}(G) + 1$ *(supondo que G é finito).*

(iii) *Os vértices u e v não são conexos em $G - a$.*

(iv) *A aresta a não pertence a nenhum circuito de G.*

Grafos particulares

Grafos completos e grafos nulos

Definição

Um grafo simples G diz-se completo quando todos os pares de vértices são adjacentes.

Grafos completos e grafos nulos

Definição

Um grafo simples G diz-se completo quando todos os pares de vértices são adjacentes. Um grafo $G = (V, E, \psi)$ diz-se nulo quando $E = \varnothing$; ou seja, quando não tem arestas.

Grafos completos e grafos nulos

Definição

Um grafo simples G diz-se completo quando todos os pares de vértices são adjacentes. Um grafo $G = (V, E, \psi)$ diz-se nulo quando $E = \varnothing$; ou seja, quando não tem arestas.

Nota

- A menos de isomorfismo, existe um único grafo completo de ordem n. Denota-se este grafo por K_n, e $\epsilon(K_n) = \binom{n}{2}$ $(n \in \mathbb{N})$.

Grafos completos e grafos nulos

Definição

Um grafo simples G diz-se completo quando todos os pares de vértices são adjacentes. Um grafo $G = (V, E, \psi)$ diz-se nulo quando $E = \varnothing$; ou seja, quando não tem arestas.

Nota

- A menos de isomorfismo, existe um único grafo completo de ordem n. Denota-se este grafo por K_n, e $\epsilon(K_n) = \binom{n}{2}$ $(n \in \mathbb{N})$.
- Cada grafo nulo é simples, de facto, os grafos nulos são precisamente os grafos complementares dos grafos completos. Portanto, denotamos o grafo nulo com n vértices por $K_n^{\complement}$.

Grafos completos e grafos nulos

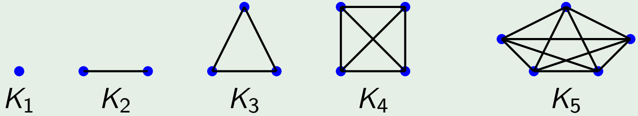
Definição

Um grafo simples G diz-se completo quando todos os pares de vértices são adjacentes. Um grafo $G = (V, E, \psi)$ diz-se nulo quando $E = \varnothing$; ou seja, quando não tem arestas.

Nota

- A menos de isomorfismo, existe um único grafo completo de ordem n. Denota-se este grafo por K_n, e $\epsilon(K_n) = \binom{n}{2}$ $(n \in \mathbb{N})$.
- Cada grafo nulo é simples, de facto, os grafos nulos são precisamente os grafos complementares dos grafos completos. Portanto, denotamos o grafo nulo com n vértices por $K_n^{\complement}$.

Exemplos (Grafos completos)



Grafos regulares

Definição

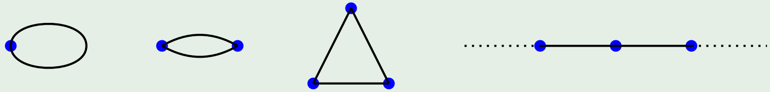
Seja $k \in \mathbb{N}$. Um grafo G diz-se *k-regular* quando todos os seus vértices têm grau k. Um grafo G diz-se *regular* quando G é k-regular para algum $k \in \mathbb{N}$.

Grafos regulares

Definição

Seja $k \in \mathbb{N}$. Um grafo G diz-se *k-regular* quando todos os seus vértices têm grau k. Um grafo G diz-se *regular* quando G é k-regular para algum $k \in \mathbb{N}$.

Exemplos (Grafos 2-regulares)

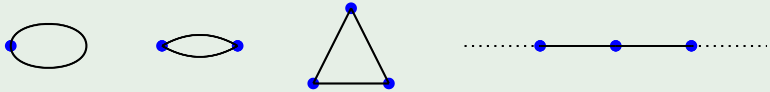


Grafos regulares

Definição

Seja $k \in \mathbb{N}$. Um grafo G diz-se k-regular quando todos os seus vértices têm grau k. Um grafo G diz-se regular quando G é k-regular para algum $k \in \mathbb{N}$.

Exemplos (Grafos 2-regulares)



Nota

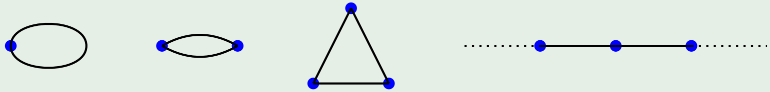
- Os grafos 3-regulares designam-se por grafos cúbicos.

Grafos regulares

Definição

Seja $k \in \mathbb{N}$. Um grafo G diz-se *k-regular* quando todos os seus vértices têm grau k. Um grafo G diz-se *regular* quando G é k-regular para algum $k \in \mathbb{N}$.

Exemplos (Grafos 2-regulares)



Nota

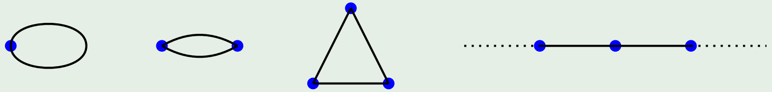
- Os grafos 3-regulares designam-se por grafos cúbicos.
- O grafo K_n é $(n-1)$-regular.

Grafos regulares

Definição

Seja $k \in \mathbb{N}$. Um grafo G diz-se *k-regular* quando todos os seus vértices têm grau k. Um grafo G diz-se *regular* quando G é k-regular para algum $k \in \mathbb{N}$.

Exemplos (Grafos 2-regulares)



Nota

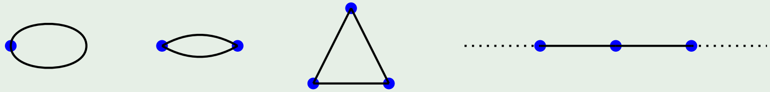
- Os grafos 3-regulares designam-se por grafos cúbicos.
- O grafo K_n é $(n-1)$-regular. De facto, um grafo simples G é $(n-1)$-regular se e só se G é completo.

Grafos regulares

Definição

Seja $k \in \mathbb{N}$. Um grafo G diz-se *k-regular* quando todos os seus vértices têm grau k. Um grafo G diz-se regular quando G é k-regular para algum $k \in \mathbb{N}$.

Exemplos (Grafos 2-regulares)



Nota

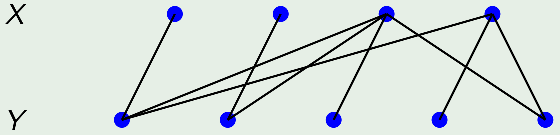
- Os grafos 3-regulares designam-se por grafos cúbicos.
- O grafo K_n é $(n-1)$-regular. De facto, um grafo simples G é $(n-1)$-regular se e só se G é completo.
- Um grafo G é 0-regular se e só se G é um grafo nulo.

Grafos bipartidos

Definição

Um grafo $G = (V, E, \psi)$ diz-se bipartido quando existem subconjuntos não-vazios $X, Y \subseteq V$ de V com $V = X \cup Y$ e $X \cap Y = \emptyset$ tais que os grafos $G[X]$ e $G[Y]$ são nulos

Exemplo

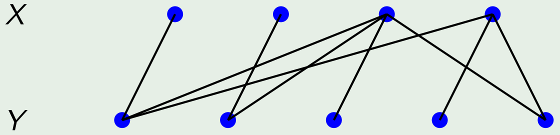


Grafos bipartidos

Definição

Um grafo $G = (V, E, \psi)$ diz-se bipartido quando existem subconjuntos não-vazios $X, Y \subseteq V$ de V com $V = X \cup Y$ e $X \cap Y = \varnothing$ tais que os grafos $G[X]$ e $G[Y]$ são nulos (isto é, não existem arestas entre qualquer par de vértices de X nem entre qualquer par de vértices de Y; ou seja, cada aresta de G tem um extremo em X e outro em Y).

Exemplo



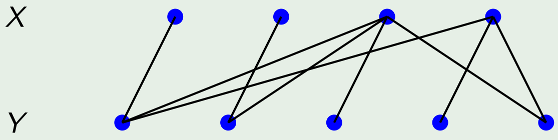
Grafos bipartidos

Definição

Um grafo $G = (V, E, \psi)$ diz-se bipartido quando existem subconjuntos não-vazios $X, Y \subseteq V$ de V com $V = X \cup Y$ e $X \cap Y = \varnothing$ tais que os grafos $G[X]$ e $G[Y]$ são nulos (isto é, não existem arestas entre qualquer par de vértices de X nem entre qualquer par de vértices de Y; ou seja, cada aresta de G tem um extremo em X e outro em Y).

Uma tal partição $\{X, Y\}$ do conjunto V dos vértices de G designa-se por bipartição dos vértices. Neste caso denota-se G por (X, Y, E, ψ) (ou simplesmente (X, Y, E) se G é simples).

Exemplo



Grafos bipartidos

Teorema

G^a *é bipartido* $\iff$ G *não tem circuitos*^{b} *de comprimento ímpar.*

[a] com pelo menos dois vértices
[b] circuito = passeio fechado sem repetição de arestas.

Grafos bipartidos

Teorema

G é bipartido $\iff$ G não tem circuitos de comprimento ímpar.

Demonstração.

$\square$

Grafos bipartidos

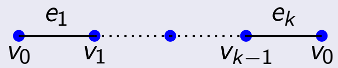
Teorema

G é bipartido $\iff$ *G não tem circuitos de comprimento ímpar.*

Demonstração.

Suponha que G é bipartido (com partição $\{X, Y\}$) e seja

P:



um circuito em G. Suponhamos que $v_0 \in X$.

Grafos bipartidos

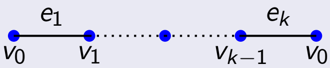
Teorema

G é bipartido $\iff$ G não tem circuitos de comprimento ímpar.

Demonstração.

Suponha que G é bipartido (com partição $\{X, Y\}$) e seja

P:



um circuito em G. Suponhamos que $v_0 \in X$. Então, $v_1 \in Y$, $v_2 \in X$, $\ldots$, $v_{k-1} \in Y$ e $v_0 \in X$.

$\square$

Grafos bipartidos

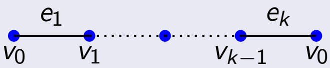
Teorema

G é bipartido $\iff$ G não tem circuitos de comprimento ímpar.

Demonstração.

Suponha que G é bipartido (com partição $\{X, Y\}$) e seja

P:



um circuito em G. Suponhamos que $v_0 \in X$. Então, $v_1 \in Y$, $v_2 \in X$, ..., $v_{k-1} \in Y$ e $v_0 \in X$. Portanto, há um número ímpar de vértices e por isso

□

Grafos bipartidos

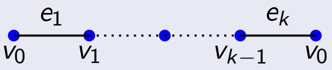
Teorema

G é bipartido $\iff$ G não tem circuitos de comprimento ímpar.

Demonstração.

Suponha que G é bipartido (com partição $\{X, Y\}$) e seja

P:



um circuito em G. Suponhamos que $v_0 \in X$. Então, $v_1 \in Y$, $v_2 \in X$, ..., $v_{k-1} \in Y$ e $v_0 \in X$. Portanto, há um número ímpar de vértices e por isso um número par de arestas.

□

Grafos bipartidos

Teorema

G é bipartido $\iff$ G não tem circuitos de comprimento ímpar.

Demonstração.

Suponha agora que $G = (V, E, \psi)$ não tem circuitos de comprimento ímpar (e G é conexo).

$\square$

Grafos bipartidos

Teorema

G é bipartido $\iff$ G não tem circuitos de comprimento ímpar.

Demonstração.

Suponha agora que $G = (V, E, \psi)$ não tem circuitos de comprimento ímpar (e G é conexo). Seja $x_0 \in V$. Consideramos

$$X = \{x \in V \mid \text{dist}(x, x_0) \text{ é par}\}, \; Y = \{y \in V \mid \text{dist}(x, x_0) \text{ é ímpar}\}.$$

$\square$

Teorema

G é bipartido $\iff$ G não tem circuitos de comprimento ímpar.

Demonstração.

Suponha agora que $G = (V, E, \psi)$ não tem circuitos de comprimento ímpar (e G é conexo). Seja $x_0 \in V$. Consideramos

$$X = \{x \in V \mid \text{dist}(x, x_0) \text{ é par}\}, \ Y = \{y \in V \mid \text{dist}(x, x_0) \text{ é ímpar}\}.$$

Suponhamos que existem $x, x' \in X$ (ou em Y) adjacentes (com $a \in E$).

$\square$

Teorema

G é bipartido $\iff$ G não tem circuitos de comprimento ímpar.

Demonstração.

Suponha agora que $G = (V, E, \psi)$ não tem circuitos de comprimento ímpar (e G é conexo). Seja $x_0 \in V$. Consideramos

$$X = \{x \in V \mid \mathrm{dist}(x, x_0) \text{ é par}\},\; Y = \{y \in V \mid \mathrm{dist}(x, x_0) \text{ é ímpar}\}.$$

Suponhamos que existem $x, x' \in X$ adjacentes (com $a \in E$). Sejam

P: 

P': 

caminhos com menor comprimento.

$\square$

Grafos bipartidos

Teorema

G é bipartido $\iff$ G não tem circuitos de comprimento ímpar.

Demonstração.

Suponha agora que $G = (V, E, \psi)$ não tem circuitos de comprimento ímpar (e G é conexo). Seja $x_0 \in V$. Consideramos

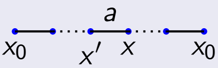
$$X = \{x \in V \mid \text{dist}(x, x_0) \text{ é par}\}, \quad Y = \{y \in V \mid \text{dist}(x, x_0) \text{ é ímpar}\}.$$

Suponhamos que existem $x, x' \in X$ adjacentes (com $a \in E$). Sejam

P: 

P': 

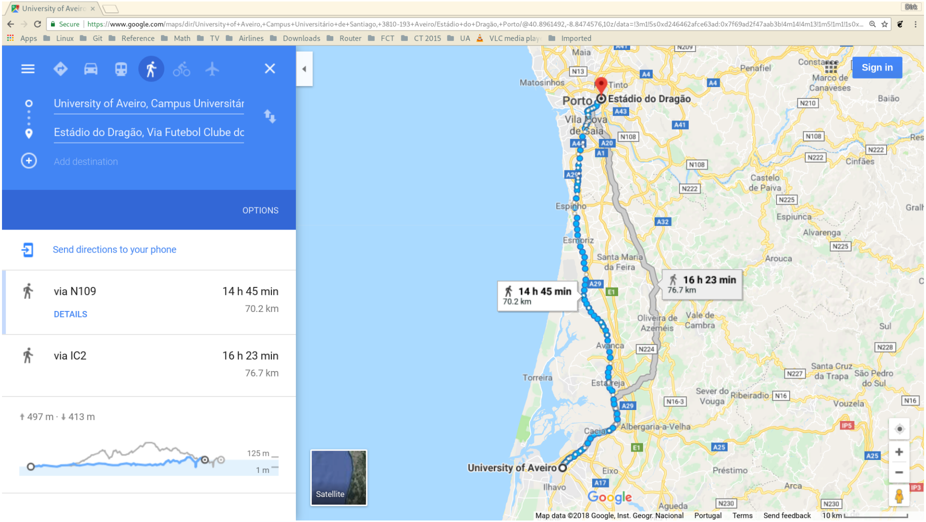
caminhos com menor comprimento. Portanto,



é um passeio fechado de comprimento ímpar, logo existe um circuito de comprimento ímpar (TPC!!), uma contradição. □

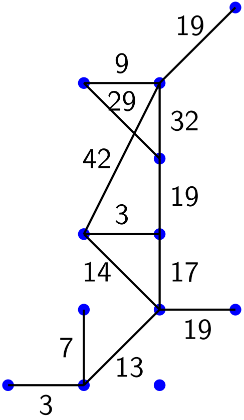
Problemas de caminho de "custo mínimo" em grafos

O problema



University of Aveiro, Campus Universitár
Estádio do Dragão, Via Futebol Clube dc
Add destination
OPTIONS
Send directions to your phone
via N109
14 h 45 min
70.2 km
DETAILS
via IC2
16 h 23 min
76.7 km
497 m · 413 m
125 m
1 m
Satellite
Porto
Estádio do Dragão
University of Aveiro
14 h 45 min
70.2 km
16 h 23 min
76.7 km
Sign in
Map data ©2018 Google, Inst. Geogr. Nacional
Portugal
Terms
Send feedback
10 km

Formalizar o problema



- vértices = cruzamentos
- arestas = estradas com distância/tempo/preço/...

Grafos com custos

Definição

Um grafo com custos não negativos nas arestas $G = (V, E, W)$ é dado por um grafos simples (V, E) e uma *matriz de custos*

$$W \colon V \times V \longrightarrow [0, \infty]$$

tais que, $W(u, v) = W(v, u)$, $W(u, u) = 0$ e, para todos os $u, v \in V$, $W(u, v) = \infty$ se $uv \notin E$.

Grafos com custos

Definição

Um grafo com custos não negativos nas arestas $G = (V, E, W)$ é dado por um grafos simples (V, E) e uma *matriz de custos*

$$W\colon V \times V \longrightarrow [0, \infty]$$

tais que, $W(u, v) = W(v, u)$, $W(u, u) = 0$ e, para todos os $u, v \in V$, $W(u, v) = \infty$ se $uv \notin E$. (Portanto, não precisamos E.)

Grafos com custos

Definição

Um grafo com custos não negativos nas arestas $G = (V, E, W)$ é dado por um grafos simples (V, E) e uma *matriz de custos*

$$W\colon V \times V \longrightarrow [0, \infty]$$

tais que, $W(u, v) = W(v, u)$, $W(u, u) = 0$ e, para todos os $u, v \in V$, $W(u, v) = \infty$ se $uv \notin E$. (Portanto, não precisamos E.)

Para cada caminho $P = (v_0, v_1, \ldots, v_k)$ em G, o custo de P é

$$W(P) = \sum_{i=0}^{k-1} W(v_i, v_{i+1})$$

(onde $\alpha + \infty = \infty = \infty + \alpha$).

Grafos com custos

Definição

Um grafo com custos não negativos nas arestas $G = (V, E, W)$ é dado por um grafos simples (V, E) e uma *matriz de custos*

$$W: V \times V \longrightarrow [0, \infty]$$

tais que, $W(u, v) = W(v, u)$, $W(u, u) = 0$ e, para todos os $u, v \in V$, $W(u, v) = \infty$ se $uv \notin E$. (Portanto, não precisamos E.)

Para cada caminho $P = (v_0, v_1, \ldots, v_k)$ em G, o custo de P é

$$W(P) = \sum_{i=0}^{k-1} W(v_i, v_{i+1})$$

(onde $\alpha + \infty = \infty = \infty + \alpha$).

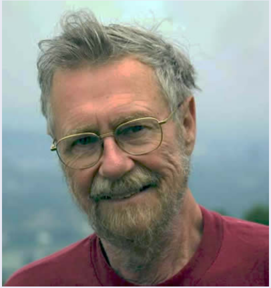
Objetivo

Encontrar o caminho *de menor custo* entre dois vértices.

O algoritmo de Dijkstra

Considerações iniciais

Se $(v_0, v_1, \ldots, v_{k-1}, v_k)$ é o caminho de "menor custo" entre v_0 e v_k, então $(v_0, v_1, \ldots, v_{k-1})$ é o caminho de "menor custo" entre v_0 e v_{k-1}.



Edsger W. Dijkstra (1959). «A note on two problems in connexion with graphs». Em: *Numerische Mathematik* **1**.(1), pp. 269–271.

Edsger Wybe Dijkstra (1930 – 2002), matemático holandês.

O algoritmo de Dijkstra

As variáveis

- `start` = o vértice inicial.

O algoritmo de Dijkstra

As variáveis

- `start` = o vértice inicial.
- Para cada $v \in V$:

O algoritmo de Dijkstra

As variáveis

- `start` $=$ o vértice inicial.
- Para cada $v \in V$:
  - **custo**(v) $=$ "custo" do caminho de menor "custo" entre s e v (até o momento).

O algoritmo de Dijkstra

As variáveis

- `start` = o vértice inicial.
- Para cada $v \in V$:
  - **custo**(v) = "custo" do caminho de menor "custo" entre s e v (até o momento).
  - **ant**(v) = antecessor de v no caminho de menor "custo" entre s e v (até o momento).

O algoritmo de Dijkstra

As variáveis

- `start` $=$ o vértice inicial.
- Para cada $v \in V$:
  - **custo**(v) $=$ "custo" do caminho de menor "custo" entre s e v (até o momento).
  - **ant**(v) $=$ antecessor de v no caminho de menor "custo" entre s e v (até o momento).
- **temp** $=$ lista dos vértices com valores temporários.

O algoritmo de Dijkstra

As variáveis

- `start` = o vértice inicial.
- Para cada $v \in V$:
  - **custo**(v) = "custo" do caminho de menor "custo" entre s e v (até o momento).
  - **ant**(v) = antecessor de v no caminho de menor "custo" entre s e v (até o momento).
- **temp** = lista dos vértices com valores temporários.
- `menor` = vértice de menor custo (neste momento).

O algoritmo de Dijkstra

O algoritmo

- Inicializar as variáveis:

O algoritmo de Dijkstra

O algoritmo

- Inicializar as variáveis:
  - Para cada $v \in V$: $\textbf{custo}(v) = \infty$, $\textbf{ant}(v) = \varnothing$.

O algoritmo de Dijkstra

O algoritmo

- Inicializar as variáveis:
  - Para cada $v \in V$: $\textbf{custo}(v) = \infty$, $\textbf{ant}(v) = \varnothing$.
  - $\textbf{custo}(\texttt{start}) = 0$.

O algoritmo de Dijkstra

O algoritmo

- Inicializar as variáveis:
  - Para cada $v \in V$: $\mathbf{custo}(v) = \infty$, $\mathbf{ant}(v) = \varnothing$.
  - $\mathbf{custo}(\texttt{start}) = 0$.
  - $\mathbf{temp} = V \setminus \{\texttt{start}\}$ e $\texttt{menor} = \texttt{start}$.

O algoritmo de Dijkstra

O algoritmo

- Inicializar as variáveis:
  - Para cada $v \in V$: $\textbf{custo}(v) = \infty$, $\textbf{ant}(v) = \varnothing$.
  - $\textbf{custo}(\texttt{start}) = 0$.
  - $\textbf{temp} = V \setminus \{\texttt{start}\}$ e $\texttt{menor} = \texttt{start}$.
- Repetir:

Até $\texttt{menor} =$ o vértice terminal.

O algoritmo de Dijkstra

O algoritmo

- Inicializar as variáveis:
  - Para cada $v \in V$: $\textbf{custo}(v) = \infty$, $\textbf{ant}(v) = \varnothing$.
  - $\textbf{custo}(\texttt{start}) = 0$.
  - $\textbf{temp} = V \setminus \{\texttt{start}\}$ e $\texttt{menor} = \texttt{start}$.
- Repetir:
  - $c_{\text{aux}} = \infty$.

Até $\texttt{menor}$ = o vértice terminal.

O algoritmo de Dijkstra

O algoritmo

- Inicializar as variáveis:
  - Para cada $v \in V$: $\textbf{custo}(v) = \infty$, $\textbf{ant}(v) = \varnothing$.
  - $\textbf{custo}(\texttt{start}) = 0$.
  - $\textbf{temp} = V \setminus \{\texttt{start}\}$ e $\texttt{menor} = \texttt{start}$.
- Repetir:
  - $c_{\text{aux}} = \infty$.
  - Para todo o v em **temp**:

Até $\texttt{menor} =$ o vértice terminal.

O algoritmo de Dijkstra

O algoritmo

- Inicializar as variáveis:
  - Para cada $v \in V$: $\textbf{custo}(v) = \infty$, $\textbf{ant}(v) = \varnothing$.
  - $\textbf{custo}(\texttt{start}) = 0$.
  - $\textbf{temp} = V \setminus \{\texttt{start}\}$ e $\texttt{menor} = \texttt{start}$.
- Repetir:
  - $c_{\text{aux}} = \infty$.
  - Para todo o v em **temp**:
    - Se $\textbf{custo}(v) > \textbf{custo}(\texttt{menor}) + W(\texttt{menor}, v)$, então $\textbf{custo}(v) = \textbf{custo}(\texttt{menor}) + W(\texttt{menor}, v)$ e $\textbf{ant}(v) = \texttt{menor}$.

Até $\texttt{menor} =$ o vértice terminal.

O algoritmo de Dijkstra

O algoritmo

- Inicializar as variáveis:
  - Para cada $v \in V$: $\textbf{custo}(v) = \infty$, $\textbf{ant}(v) = \varnothing$.
  - $\textbf{custo}(\texttt{start}) = 0$.
  - $\textbf{temp} = V \setminus \{\texttt{start}\}$ e $\texttt{menor} = \texttt{start}$.
- Repetir:
  - $c_{\text{aux}} = \infty$.
  - Para todo o v em **temp**:
    - Se $\textbf{custo}(v) > \textbf{custo}(\texttt{menor}) + W(\texttt{menor}, v)$, então $\textbf{custo}(v) = \textbf{custo}(\texttt{menor}) + W(\texttt{menor}, v)$ e $\textbf{ant}(v) = \texttt{menor}$.
    - Se $\textbf{custo}(v) < c_{\text{aux}}$ então $c_{\text{aux}} = \textbf{custo}(v)$ e $x = v$ (lembrar do "menor custo").

Até $\texttt{menor} =$ o vértice terminal.

O algoritmo de Dijkstra

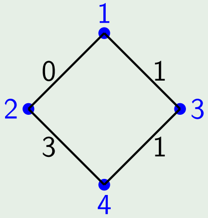
O algoritmo

- Inicializar as variáveis:
  - Para cada $v \in V$: $\textbf{custo}(v) = \infty$, $\textbf{ant}(v) = \varnothing$.
  - $\textbf{custo}(\texttt{start}) = 0$.
  - $\textbf{temp} = V \setminus \{\texttt{start}\}$ e $\texttt{menor} = \texttt{start}$.
- Repetir:
  - $c_{\mathrm{aux}} = \infty$.
  - Para todo o v em **temp**:
    - Se $\textbf{custo}(v) > \textbf{custo}(\texttt{menor}) + W(\texttt{menor}, v)$, então $\textbf{custo}(v) = \textbf{custo}(\texttt{menor}) + W(\texttt{menor}, v)$ e $\textbf{ant}(v) = \texttt{menor}$.
    - Se $\textbf{custo}(v) < c_{\mathrm{aux}}$ então $c_{\mathrm{aux}} = \textbf{custo}(v)$ e $x = v$ (lembrar do "menor custo").
  - $\textbf{temp} = \textbf{temp} \setminus \{x\}$ e $\texttt{menor} = x$.

  Até $\texttt{menor} =$ o vértice terminal.

O algoritmo de Dijkstra

Exemplo



- vértice inicial: 1.
- vértice terminal: 4.

O algoritmo de Dijkstra

Exemplo

- Inicializar: $\textbf{custo}(2) = \textbf{custo}(3) = \textbf{custo}(4) = \infty$, $\textbf{ant}(1) = \textbf{ant}(2) = \textbf{ant}(3) = \textbf{ant}(4) = \varnothing$, $\textbf{temp} = \{2, 3, 4\}$, $\texttt{menor} = 1$, $\textbf{custo}(1) = 0$.

O algoritmo de Dijkstra

Exemplo

- Inicializar: $\textbf{custo}(2) = \textbf{custo}(3) = \textbf{custo}(4) = \infty$, $\textbf{ant}(1) = \textbf{ant}(2) = \textbf{ant}(3) = \textbf{ant}(4) = \varnothing$, $\textbf{temp} = \{2, 3, 4\}$, $\texttt{menor} = 1$, $\textbf{custo}(1) = 0$.
- Iteração 1: $c_{\text{aux}} = \infty$ $(\texttt{menor} = 1)$ $\textbf{custo}(2) = 0$, $\textbf{ant}(2) = 1$, $c_{\text{aux}} = 0$, $x = 2$, $\textbf{custo}(3) = 1$, $\textbf{ant}(3) = 1$,

O algoritmo de Dijkstra

Exemplo

- Inicializar: $\textbf{custo}(2) = \textbf{custo}(3) = \textbf{custo}(4) = \infty$, $\textbf{ant}(1) = \textbf{ant}(2) = \textbf{ant}(3) = \textbf{ant}(4) = \varnothing$, $\textbf{temp} = \{2, 3, 4\}$, $\texttt{menor} = 1$, $\textbf{custo}(1) = 0$.
- Iteração 1: $c_{\mathrm{aux}} = \infty$ $(\texttt{menor} = 1)$
  $\textbf{custo}(2) = 0$, $\textbf{ant}(2) = 1$, $c_{\mathrm{aux}} = 0$, $x = 2$,
  $\textbf{custo}(3) = 1$, $\textbf{ant}(3) = 1$,
  $\textbf{temp} = \{3, 4\}$, $\texttt{menor} = 2$.

O algoritmo de Dijkstra

Exemplo

- Inicializar: $\textbf{custo}(2) = \textbf{custo}(3) = \textbf{custo}(4) = \infty$, $\textbf{ant}(1) = \textbf{ant}(2) = \textbf{ant}(3) = \textbf{ant}(4) = \varnothing$, $\textbf{temp} = \{2, 3, 4\}$, $\texttt{menor} = 1$, $\textbf{custo}(1) = 0$.
- Iteração 1: $c_{\mathrm{aux}} = \infty$ $(\texttt{menor} = 1)$
  $\textbf{custo}(2) = 0$, $\textbf{ant}(2) = 1$, $c_{\mathrm{aux}} = 0$, $x = 2$,
  $\textbf{custo}(3) = 1$, $\textbf{ant}(3) = 1$,
  $\textbf{temp} = \{3, 4\}$, $\texttt{menor} = 2$.
- Iteração 2: $c_{\mathrm{aux}} = \infty$ $(\texttt{menor} = 2)$
  $\textbf{custo}(4) = 3$, $\textbf{ant}(4) = 2$, $c_{\mathrm{aux}} = 3$, $x = 4$,
  $\textbf{custo}(3) = 1$, $\textbf{ant}(3) = 1$, $c_{\mathrm{aux}} = 1$, $x = 3$,

O algoritmo de Dijkstra

Exemplo

- Inicializar: $\textbf{custo}(2) = \textbf{custo}(3) = \textbf{custo}(4) = \infty$, $\textbf{ant}(1) = \textbf{ant}(2) = \textbf{ant}(3) = \textbf{ant}(4) = \varnothing$, $\textbf{temp} = \{2, 3, 4\}$, $\texttt{menor} = 1$, $\textbf{custo}(1) = 0$.
- Iteração 1: $c_{\mathrm{aux}} = \infty$ $(\texttt{menor} = 1)$
  $\textbf{custo}(2) = 0$, $\textbf{ant}(2) = 1$, $c_{\mathrm{aux}} = 0$, $x = 2$,
  $\textbf{custo}(3) = 1$, $\textbf{ant}(3) = 1$,
  $\textbf{temp} = \{3, 4\}$, $\texttt{menor} = 2$.
- Iteração 2: $c_{\mathrm{aux}} = \infty$ $(\texttt{menor} = 2)$
  $\textbf{custo}(4) = 3$, $\textbf{ant}(4) = 2$, $c_{\mathrm{aux}} = 3$, $x = 4$,
  $\textbf{custo}(3) = 1$, $\textbf{ant}(3) = 1$, $c_{\mathrm{aux}} = 1$, $x = 3$,
  $\textbf{temp} = \{4\}$, $\texttt{menor} = 3$.

O algoritmo de Dijkstra

Exemplo

- Inicializar: $\textbf{custo}(2) = \textbf{custo}(3) = \textbf{custo}(4) = \infty$,
  $\textbf{ant}(1) = \textbf{ant}(2) = \textbf{ant}(3) = \textbf{ant}(4) = \varnothing$,
  $\textbf{temp} = \{2, 3, 4\}$, $\texttt{menor} = 1$, $\textbf{custo}(1) = 0$.
- Iteração 1: $c_{\mathrm{aux}} = \infty$ $(\texttt{menor} = 1)$
  $\textbf{custo}(2) = 0$, $\textbf{ant}(2) = 1$, $c_{\mathrm{aux}} = 0$, $x = 2$,
  $\textbf{custo}(3) = 1$, $\textbf{ant}(3) = 1$,
  $\textbf{temp} = \{3, 4\}$, $\texttt{menor} = 2$.
- Iteração 2: $c_{\mathrm{aux}} = \infty$ $(\texttt{menor} = 2)$
  $\textbf{custo}(4) = 3$, $\textbf{ant}(4) = 2$, $c_{\mathrm{aux}} = 3$, $x = 4$,
  $\textbf{custo}(3) = 1$, $\textbf{ant}(3) = 1$, $c_{\mathrm{aux}} = 1$, $x = 3$,
  $\textbf{temp} = \{4\}$, $\texttt{menor} = 3$.
- Iteração 3: $c_{\mathrm{aux}} = \infty$
  $\textbf{custo}(4) = 2$, $\textbf{ant}(4) = 3$, $c_{\mathrm{aux}} = 2$, $x = 4$,

O algoritmo de Dijkstra

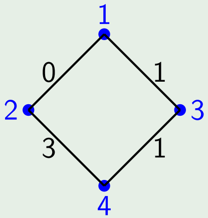
Exemplo

- Inicializar: $\textbf{custo}(2) = \textbf{custo}(3) = \textbf{custo}(4) = \infty$, $\textbf{ant}(1) = \textbf{ant}(2) = \textbf{ant}(3) = \textbf{ant}(4) = \varnothing$, $\textbf{temp} = \{2, 3, 4\}$, $\texttt{menor} = 1$, $\textbf{custo}(1) = 0$.
- Iteração 1: $c_{\text{aux}} = \infty$ $(\texttt{menor} = 1)$
  $\textbf{custo}(2) = 0$, $\textbf{ant}(2) = 1$, $c_{\text{aux}} = 0$, $x = 2$,
  $\textbf{custo}(3) = 1$, $\textbf{ant}(3) = 1$,
  $\textbf{temp} = \{3, 4\}$, $\texttt{menor} = 2$.
- Iteração 2: $c_{\text{aux}} = \infty$ $(\texttt{menor} = 2)$
  $\textbf{custo}(4) = 3$, $\textbf{ant}(4) = 2$, $c_{\text{aux}} = 3$, $x = 4$,
  $\textbf{custo}(3) = 1$, $\textbf{ant}(3) = 1$, $c_{\text{aux}} = 1$, $x = 3$,
  $\textbf{temp} = \{4\}$, $\texttt{menor} = 3$.
- Iteração 3: $c_{\text{aux}} = \infty$
  $\textbf{custo}(4) = 2$, $\textbf{ant}(4) = 3$, $c_{\text{aux}} = 2$, $x = 4$,
  $\textbf{temp} = \varnothing$, $\texttt{menor} = 4$.

O algoritmo de Dijkstra

Exemplo

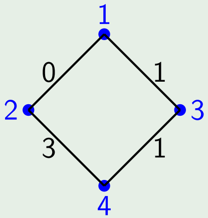
1	2	3	4	`menor`	**temp**
$(0, -)$	$(\infty, -)$	$(\infty, -)$	$(\infty, -)$	1	$\{2, 3, 4\}$



O algoritmo de Dijkstra

Exemplo

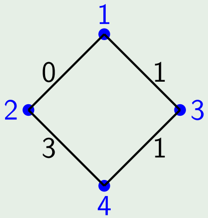
1	2	3	4	`menor`	**temp**
$(0,-)$	$(\infty,-)$	$(\infty,-)$	$(\infty,-)$	1	$\{2,3,4\}$
	$(0,1)$	$(1,1)$	$(\infty,-)$	2	$\{3,4\}$



O algoritmo de Dijkstra

Exemplo

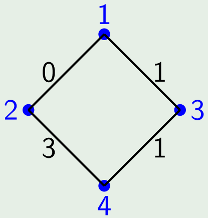
1	2	3	4	`menor`	**temp**
$(0, -)$	$(\infty, -)$	$(\infty, -)$	$(\infty, -)$	1	$\{2, 3, 4\}$
	$(0, 1)$	$(1, 1)$	$(\infty, -)$	2	$\{3, 4\}$
		$(1, 1)$	$(3, 2)$	3	$\{4\}$



O algoritmo de Dijkstra

Exemplo

1	2	3	4	`menor`	**temp**
$(0,-)$	$(\infty,-)$	$(\infty,-)$	$(\infty,-)$	1	$\{2,3,4\}$
	$(0,1)$	$(1,1)$	$(\infty,-)$	2	$\{3,4\}$
		$(1,1)$	$(3,2)$	3	$\{4\}$
			$(2,3)$	4	$\varnothing$



Simon Peyton Jones e Andrew Goldberg (2010). «Getting from A to B: fast route-finding on slow computers». URL: https://www.microsoft.com/en-us/research/video/getting-from-a-b-fast-route-finding-using-slow-computers/. A talk by Simon Peyton-Jones for Think Computer Science 2010.

Stephen Dolan (2013). «Fun with semirings: a functional pearl on the abuse of linear algebra». Em: *Proceedings of the 18th ACM SIGPLAN international conference on Functional programming - ICFP '13*. Vol. 48. 9. ACM. ACM Press, pp. 101–110. URL: https://www.cl.cam.ac.uk/~sd601/papers/semirings.pdf.