Algoritmos e Teoria dos Grafos

Tópico 10: Trilhas e Grafos Eulerianos

Renato Carmo
André Guedes
Murilo Silva
Nicollas Sdroievski

Departamento de Informática da UFPR

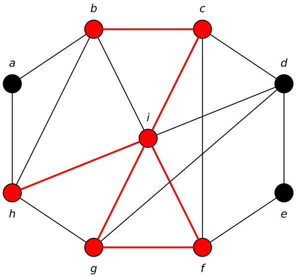
2026 - Primeiro semestre

Aula de hoje

Aula de hoje

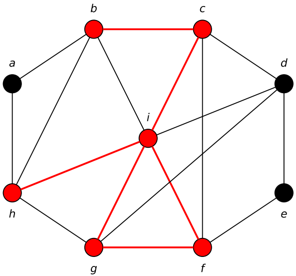
1. Definição
2. Trilhas e Grafos Eulerianos
3. Algoritmos

Definição



Trilha (b, c, i, g, f, i, h)

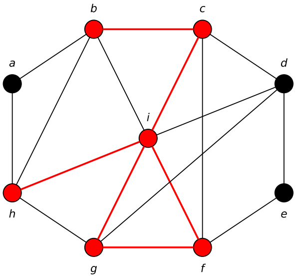
Definição



Trilha (b, c, i, g, f, i, h)

trilha

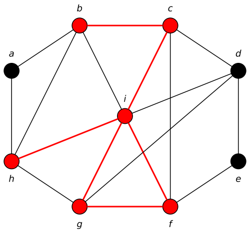
Definição



Trilha (b, c, i, g, f, i, h)

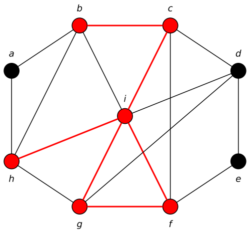
trilha: passeio cujas arestas são todas distintas

Teorema 56



Trilha (b, c, i, g, f, i, h)

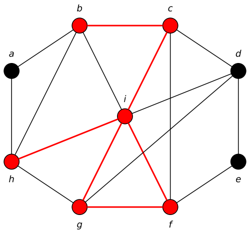
Teorema 56



Trilha (b, c, i, g, f, i, h)

vértice interno da trilha T

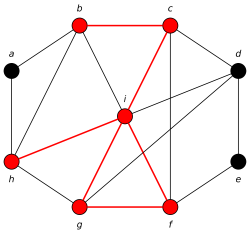
Teorema 56



Trilha (b, c, i, g, f, i, h)

vértice interno da trilha T tem **grau par** em $G[T]$

Teorema 56



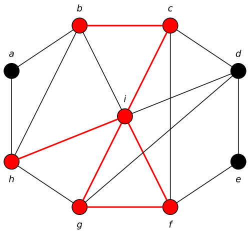
Trilha (b, c, i, g, f, i, h)

vértice interno da trilha T tem **grau par** em $G[T]$

Demonstração.

1. $T := (v_0, \ldots, v_n)$

Teorema 56



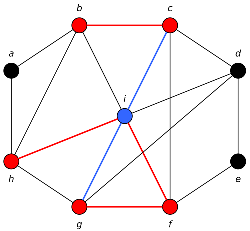
Trilha (b, c, i, g, f, i, h)

vértice interno da trilha T tem **grau par** em $G[T]$

Demonstração.

1. $T := (v_0, \ldots, v_n)$: trilha

Teorema 56



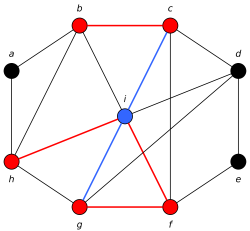
Trilha (b, c, i, g, f, i, h)
$v_3 = i$

vértice interno da trilha T tem **grau par** em $G[T]$

Demonstração.

1. $T := (v_0, \ldots, v_n)$: trilha
2. $k \in [1..n-1]$

Teorema 56



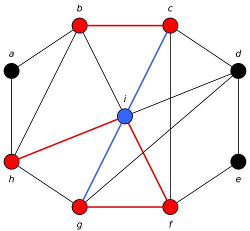
Trilha (b, c, i, g, f, i, h)
$v_3 = i$

vértice interno da trilha T tem **grau par** em $G[T]$

Demonstração.

1. $T := (v_0, \ldots, v_n)$: trilha
2. $k \in [1..n-1]$
3. arestas de (v_{k-1}, v_k, v_{k+1}) em $\partial_G(v_k)$

Teorema 56



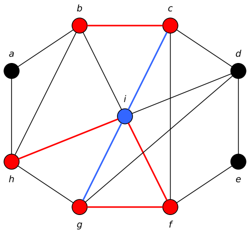
Trilha (b, c, i, g, f, i, h)
$v_3 = i$

vértice interno da trilha T tem **grau par** em $G[T]$

Demonstração.

1. $T := (v_0, \ldots, v_n)$: trilha
2. $k \in [1..n-1]$
3. arestas de (v_{k-1}, v_k, v_{k+1}) em $\partial_G(v_k)$: $\{v_{k-1}, v_k\}$ e $\{v_k, v_{k+1}\}$

Teorema 56



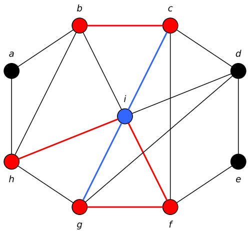
Trilha (b, c, i, g, f, i, h)
$v_3 = i$

vértice interno da trilha T tem **grau par** em $G[T]$

Demonstração.

1. $T := (v_0, \ldots, v_n)$: trilha
2. $k \in [1..n-1]$
3. arestas de (v_{k-1}, v_k, v_{k+1}) em $\partial_G(v_k)$: $\{v_{k-1}, v_k\}$ e $\{v_k, v_{k+1}\}$
4. v ocorre l vezes como vértice interno em T

Teorema 56



Trilha (b, c, i, g, f, i, h)
$v_3 = i$

vértice interno da trilha T tem **grau par** em $G[T]$

Demonstração.

1. $T := (v_0, \ldots, v_n)$: trilha
2. $k \in [1..n-1]$
3. arestas de (v_{k-1}, v_k, v_{k+1}) em $\partial_G(v_k)$: $\{v_{k-1}, v_k\}$ e $\{v_k, v_{k+1}\}$
4. v ocorre l vezes como vértice interno em $T \implies \delta_{G[T]}(v) = 2l$

$\square$

Corolário 57

Corolário 57

T: trilha fechada sobre G

Corolário 57

T: trilha fechada sobre $G \implies$ grau de todo vértice em $G[T]$ é par

Aula de hoje

1. Definição
2. Trilhas e Grafos Eulerianos
3. Algoritmos

Trilha Euleriana

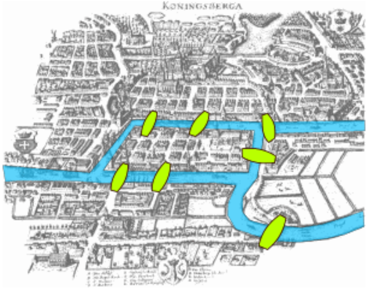
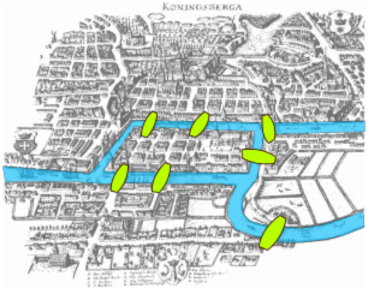


Figura: As sete pontes de Köenigsberg sobre o rio Pregel circa 1736

Trilha Euleriana



trilha euleriana

Figura: As sete pontes de Köenigsberg sobre o rio Pregel circa 1736

Trilha Euleriana

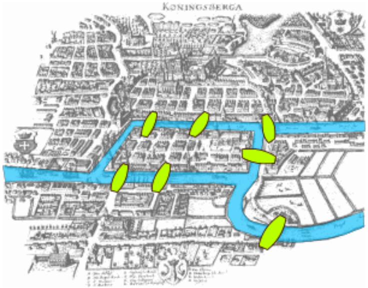


Figura: As sete pontes de Köenigsberg sobre o rio Pregel circa 1736

trilha euleriana: trilha que passa por todas as arestas

Trilha Euleriana

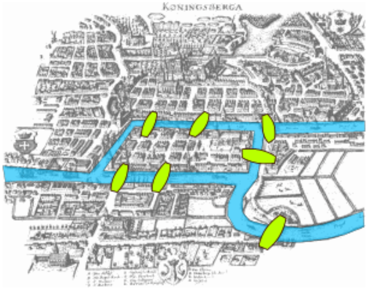


Figura: As sete pontes de Köenigsberg sobre o rio Pregel circa 1736

trilha euleriana: trilha que passa por todas as arestas

grafo euleriano

Trilha Euleriana

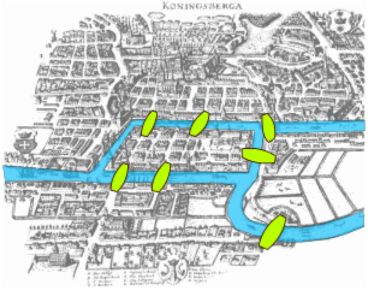


Figura: As sete pontes de Köenigsberg sobre o rio Pregel circa 1736

trilha euleriana: trilha que passa por todas as arestas
grafo euleriano: tem trilha euleriana fechada

Trilha Euleriana

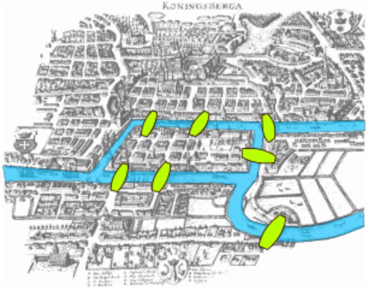


Figura: As sete pontes de Köenigsberg sobre o rio Pregel circa 1736

trilha euleriana: trilha que passa por todas as arestas
grafo euleriano: tem trilha euleriana fechada ou é trivial

Corolário 58

grafo é euleriano

Corolário 58

grafo é euleriano $\implies$ todos os vértices tem grau par

Aula de hoje

Algoritmo TrilhaFechada

TrilhaFechada(G, v)

Algoritmo TrilhaFechada

TrilhaFechada(G, v)

Entrada:

Algoritmo TrilhaFechada

TrilhaFechada(G, v)

Entrada: um grafo G cujos vértices tem grau par e um vértice v de grau positivo em G

Algoritmo TrilhaFechada

TrilhaFechada(G, v)

Entrada: um grafo G cujos vértices tem grau par e um vértice v de grau positivo em G

Saída :

Algoritmo TrilhaFechada

TrilhaFechada(G, v)

Entrada: um grafo G cujos vértices tem grau par e um vértice v de grau positivo em G

Saída : uma trilha fechada não trivial em G a partir de v

Algoritmo TrilhaFechada

TrilhaFechada(G, v)

Entrada: um grafo G cujos vértices tem grau par e um vértice v de grau positivo em G

Saída : uma trilha fechada não trivial em G a partir de v

$T \leftarrow (v)$

Algoritmo TrilhaFechada

TrilhaFechada(G, v)

Entrada: um grafo G cujos vértices tem grau par e um vértice v de grau positivo em G

Saída : uma trilha fechada não trivial em G a partir de v

$T \leftarrow (v)$

$x \leftarrow v$

Algoritmo TrilhaFechada

TrilhaFechada(G, v)

Entrada: um grafo G cujos vértices tem grau par e um vértice v de grau positivo em G

Saída : uma trilha fechada não trivial em G a partir de v

$T \leftarrow (v)$

$x \leftarrow v$

Repita

Algoritmo TrilhaFechada

TrilhaFechada(G, v)

Entrada: um grafo G cujos vértices tem grau par e um vértice v de grau positivo em G

Saída : uma trilha fechada não trivial em G a partir de v

$T \leftarrow (v)$

$x \leftarrow v$

Repita

$\quad \{x, y\} \leftarrow$ aresta de $\partial_G(x) - E(T)$

Algoritmo TrilhaFechada

TrilhaFechada(G, v)

Entrada: um grafo G cujos vértices tem grau par e um vértice v de grau positivo em G

Saída : uma trilha fechada não trivial em G a partir de v

$T \leftarrow (v)$

$x \leftarrow v$

Repita

  $\{x, y\} \leftarrow$ aresta de $\partial_G(x) - E(T)$

  acrescente y ao final de T

Algoritmo TrilhaFechada

TrilhaFechada(G, v)

Entrada: um grafo G cujos vértices tem grau par e um vértice v de grau positivo em G

Saída : uma trilha fechada não trivial em G a partir de v

$T \leftarrow (v)$

$x \leftarrow v$

Repita

    $\{x, y\} \leftarrow$ aresta de $\partial_G(x) - E(T)$

    acrescente y ao final de T

    $x \leftarrow y$

Algoritmo TrilhaFechada

TrilhaFechada(G, v)

Entrada: um grafo G cujos vértices tem grau par e um vértice v de grau positivo em G

Saída : uma trilha fechada não trivial em G a partir de v

$T \leftarrow (v)$

$x \leftarrow v$

Repita

  $\{x, y\} \leftarrow$ aresta de $\partial_G(x) - E(T)$

  acrescente y ao final de T

  $x \leftarrow y$

até que $y = v$

Algoritmo TrilhaFechada

TrilhaFechada(G, v)

Entrada: um grafo G cujos vértices tem grau par e um vértice v de grau positivo em G

Saída : uma trilha fechada não trivial em G a partir de v

$T \leftarrow (v)$

$x \leftarrow v$

Repita

  $\{x, y\} \leftarrow$ aresta de $\partial_G(x) - E(T)$

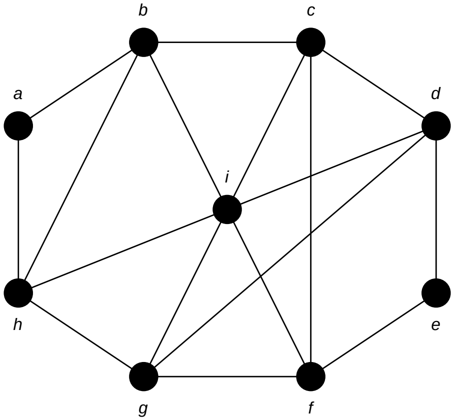
  acrescente y ao final de T

  $x \leftarrow y$

até que $y = v$

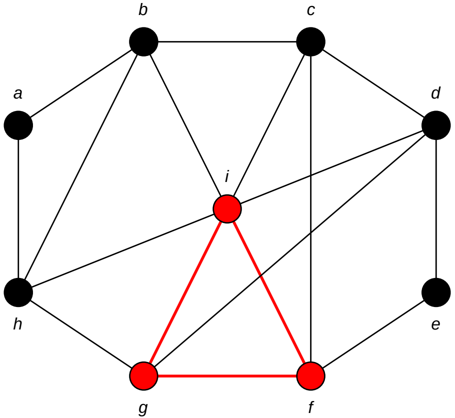
Devolva T

Execução do algoritmo



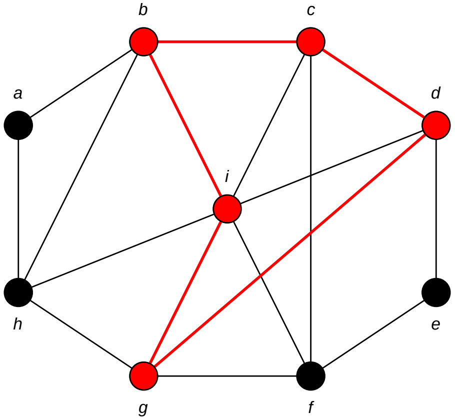
Grafo original

Execução do algoritmo



Trilha fechada (i, f, g, i)

Execução do algoritmo



Trilha fechada (i, b, c, d, g, i)

Análise do Algoritmo

Teorema 59

Se G é um grafo cujos vértices todos tem grau par e v é um vértice de grau positivo em G, então TrilhaFechada(G, v) devolve uma trilha fechada não trivial em G com início em v.

Prova do Teorema 59

- Se o algoritmo termina, então retorna uma trilha fechada não trivial em G iniciando em v

Prova do Teorema 59

- Se o algoritmo termina, então retorna uma trilha fechada não trivial em G iniciando em v
- O algoritmo termina pois

Prova do Teorema 59

- Se o algoritmo termina, então retorna uma trilha fechada não trivial em G iniciando em v
- O algoritmo termina pois
    1. a cada iteração o conjunto $E(G) - E(T)$ diminui em uma aresta,

Prova do Teorema 59

- Se o algoritmo termina, então retorna uma trilha fechada não trivial em G iniciando em v
- O algoritmo termina pois
  1. a cada iteração o conjunto $E(G) - E(T)$ diminui em uma aresta,
  2. existe uma aresta com ponta v em $E(G) - E(T)$,

Prova do Teorema 59

- Se o algoritmo termina, então retorna uma trilha fechada não trivial em G iniciando em v
- O algoritmo termina pois
  1. a cada iteração o conjunto $E(G) - E(T)$ diminui em uma aresta,
  2. existe uma aresta com ponta v em $E(G) - E(T)$,

Prova do Teorema 59

- Se o algoritmo termina, então retorna uma trilha fechada não trivial em G iniciando em v
- O algoritmo termina pois
  1. a cada iteração o conjunto $E(G) - E(T)$ diminui em uma aresta,
  2. existe uma aresta com ponta v em $E(G) - E(T)$,

  e sempre é possível escolher $\{x, y\}$ no laço. Isso porque:
  1. todos os vértices de G tem grau par em G;

Prova do Teorema 59

- Se o algoritmo termina, então retorna uma trilha fechada não trivial em G iniciando em v
- O algoritmo termina pois
  1. a cada iteração o conjunto $E(G) - E(T)$ diminui em uma aresta,
  2. existe uma aresta com ponta v em $E(G) - E(T)$,

  e sempre é possível escolher $\{x, y\}$ no laço. Isso porque:
  1. todos os vértices de G tem grau par em G;
  2. se $v = x$, então x tem grau positivo em G e $E(T) = \emptyset$, logo $\partial_G(x) - E(T) \neq \emptyset$;

Prova do Teorema 59

- Se o algoritmo termina, então retorna uma trilha fechada não trivial em G iniciando em v
- O algoritmo termina pois
  1. a cada iteração o conjunto $E(G) - E(T)$ diminui em uma aresta,
  2. existe uma aresta com ponta v em $E(G) - E(T)$,

  e sempre é possível escolher $\{x, y\}$ no laço. Isso porque:
  1. todos os vértices de G tem grau par em G;
  2. se $v = x$, então x tem grau positivo em G e $E(T) = \emptyset$, logo $\partial_G(x) - E(T) \neq \emptyset$;
  3. se $v \neq x$, então x tem grau ímpar em $G[T]$ e como $\delta_G(v) = |\partial_G(x)|$ é par, então $\partial_G(x) - E(T) \neq \emptyset$.

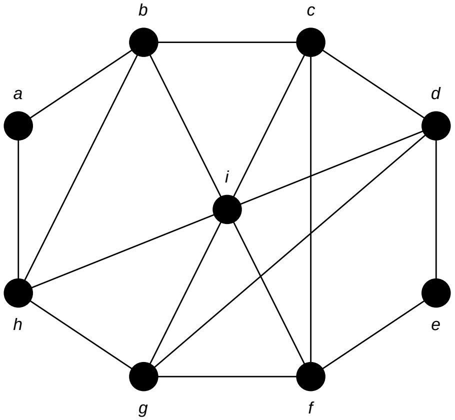
Algoritmo TrilhaEuleriana

TrilhaEuleriana(G)

Entrada: um grafo conexo G cujos vértices tem grau par
Saída : uma trilha euleriana em G

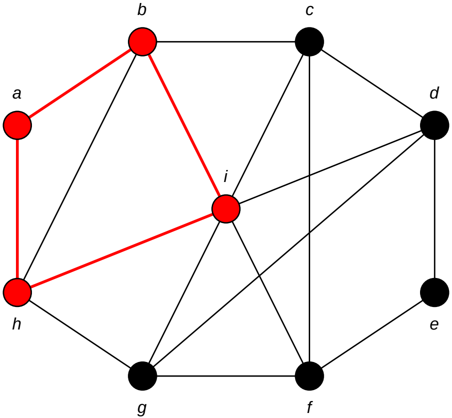
$r \leftarrow$ vértice de G
$T \leftarrow (r)$
Enquanto $E(G) \neq E(T)$
    $v \leftarrow$ vértice de T de grau positivo em $G - E(T)$
    faça uma rotação em T de forma a fazer de v seu início
    $T \leftarrow T \cdot \text{TrilhaFechada}(G - E(T), v)$
Devolva T

Execução do Algoritmo



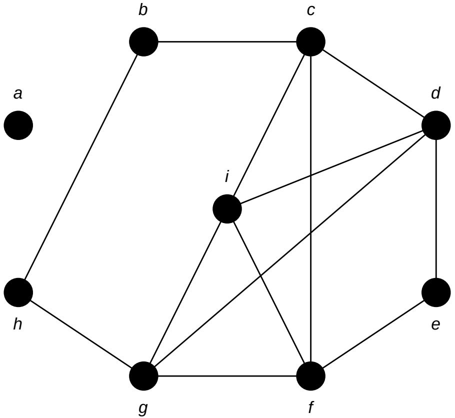
Grafo original

Execução do Algoritmo



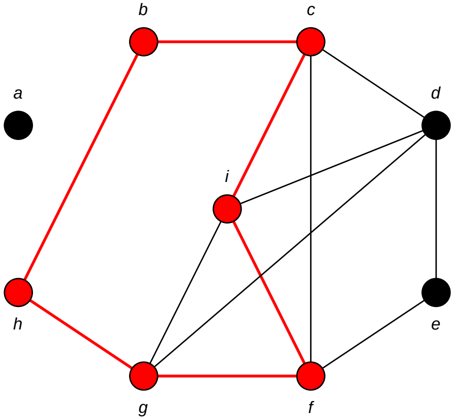
$$(a, b, i, h, a)$$

Execução do Algoritmo



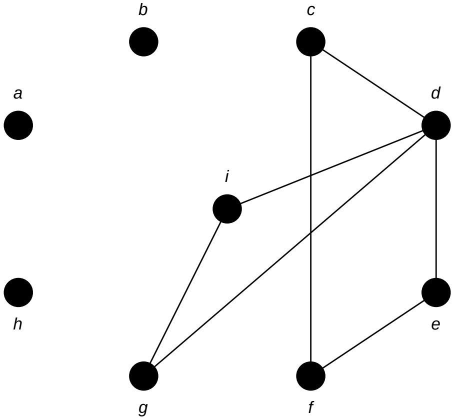
$$(a, b, i, h, a) \to (b, i, h, a, b)$$

Execução do Algoritmo



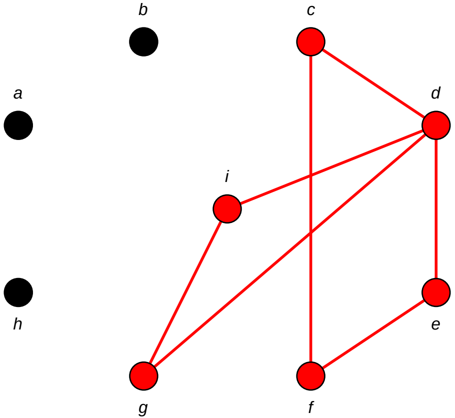
$$(b, i, h, a, b)(b, c, i, f, g, h, b)$$

Execução do Algoritmo



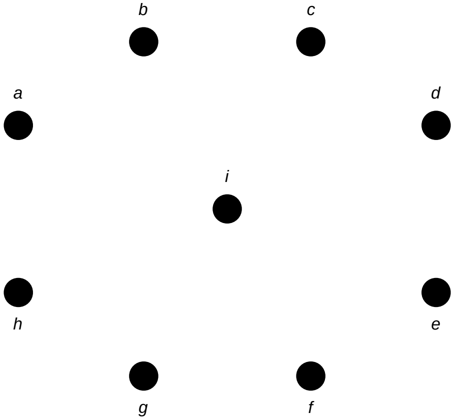
$$(b, i, h, a, b, c, i, f, g, h, b) \rightarrow (i, f, g, h, b, i, h, a, b, c, i)$$

Execução do Algoritmo



$$(i, f, g, h, b, i, h, a, b, c, i)(i, d, e, f, c, d, g, i)$$

Execução do Algoritmo



Trilha euleriana: $(i, f, g, h, b, i, h, a, b, c, i, d, e, f, c, d, g, i)$

Análise do Algoritmo

Teorema 60

Se G é um grafo conexo cujos vértices todos tem grau par, então TrilhaEuleriana(G) é uma trilha euleriana em G.

Análise do Algoritmo

Teorema 60

Se G é um grafo conexo cujos vértices todos tem grau par, então TrilhaEuleriana(G) é uma trilha euleriana em G.

Demonstração.

Basta observar que

Análise do Algoritmo

Teorema 60

Se G é um grafo conexo cujos vértices todos tem grau par, então TrilhaEuleriana(G) é uma trilha euleriana em G.

Demonstração.

Basta observar que

1. sempre é possível escolher v porque G é conexo e $E(G) - E(T) \neq \emptyset$;

Análise do Algoritmo

Teorema 60

Se G é um grafo conexo cujos vértices todos tem grau par, então $\text{TrilhaEuleriana}(G)$ é uma trilha euleriana em G.

Demonstração.

Basta observar que

1. sempre é possível escolher v porque G é conexo e $E(G) - E(T) \neq \emptyset$;
2. $G - E(T)$ é um grafo cujos vértices tem grau par e v tem grau positivo em $G - E(T)$; consequentemente (T. 59), $\text{TrilhaFechada}(G - E(T), v)$ devolve uma trilha fechada não trivial em $G - E(T)$ a partir de v;

Análise do Algoritmo

Teorema 60

Se G é um grafo conexo cujos vértices todos tem grau par, então TrilhaEuleriana(G) é uma trilha euleriana em G.

Demonstração.

Basta observar que

1. sempre é possível escolher v porque G é conexo e $E(G) - E(T) \neq \emptyset$;
2. $G - E(T)$ é um grafo cujos vértices tem grau par e v tem grau positivo em $G - E(T)$; consequentemente (T. 59), TrilhaFechada$(G - E(T), v)$ devolve uma trilha fechada não trivial em $G - E(T)$ a partir de v;
3. o laço interno termina porque $|E(T)|$ aumenta a cada iteração.

Análise do Algoritmo

Teorema 60

Se G é um grafo conexo cujos vértices todos tem grau par, então TrilhaEuleriana(G) é uma trilha euleriana em G.

Demonstração.

Basta observar que

1. sempre é possível escolher v porque G é conexo e $E(G) - E(T) \neq \emptyset$;
2. $G - E(T)$ é um grafo cujos vértices tem grau par e v tem grau positivo em $G - E(T)$; consequentemente (T. 59), TrilhaFechada($G - E(T), v$) devolve uma trilha fechada não trivial em $G - E(T)$ a partir de v;
3. o laço interno termina porque $|E(T)|$ aumenta a cada iteração.

Análise do Algoritmo

Teorema 60

Se G é um grafo conexo cujos vértices todos tem grau par, então TrilhaEuleriana(G) é uma trilha euleriana em G.

Demonstração.

Basta observar que

1. sempre é possível escolher v porque G é conexo e $E(G) - E(T) \neq \emptyset$;
2. $G - E(T)$ é um grafo cujos vértices tem grau par e v tem grau positivo em $G - E(T)$; consequentemente (T. 59), TrilhaFechada$(G - E(T), v)$ devolve uma trilha fechada não trivial em $G - E(T)$ a partir de v;
3. o laço interno termina porque $|E(T)|$ aumenta a cada iteração.

□

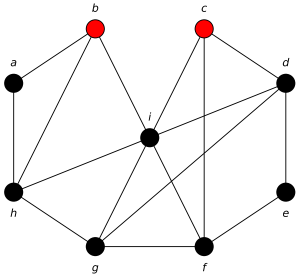
Corolário 61

grafo conexo cujos vértices todos tem grau par é euleriano

Teorema 62 (Teorema de Euler; Hierholzer, 1873)

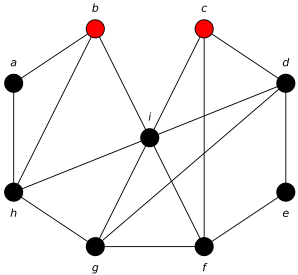
grafo conexo é euleriano se e somente se todos seus vértices tem grau par

Corolário 63



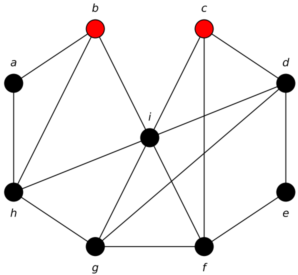
grafo conexo tem trilha euleriana aberta

Corolário 63



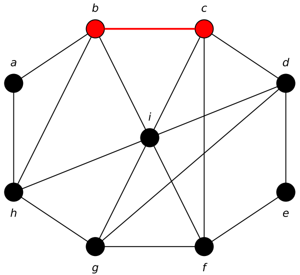
grafo conexo tem trilha euleriana aberta
se e somente se

Corolário 63



grafo conexo tem trilha euleriana aberta
se e somente se
tem exatamente dois vértices de grau ímpar

Corolário 63



grafo conexo tem trilha euleriana aberta
se e somente se
tem exatamente dois vértices de grau ímpar

Demonstração.

Exercício 82 □