UFPA

Caminho/Percurso em grafos

November 15, 2020

Adjacências

Seja um grafo $G = (V, A)$:

Adjacência de Vértices Dois vértices $v_i, v_j \in G$ são adjacentes (ou vizinhos) se existe uma aresta que liga v_i a v_j ou vice-versa.
Obs.: Aplicada a grafos não orientados.

Adjacências

Seja um grafo $G = (V, A)$:

Adjacência de Vértices Dois vértices $v_i, v_j \in G$ são adjacentes (ou vizinhos) se existe uma aresta que liga v_i a v_j ou vice-versa.
Obs.: Aplicada a grafos não orientados.

Adjacência de Arestas Duas arestas $a_i, a_j \in G$ são adjacentes se compartilham um mesmo vértice.

Sucessores e Antecessores

Seja um grafo $G = (V, A)$ e dois vértices $v_i, v_j \in G$

Sucessor v_j é sucessor de v_i se existe pelo menos um arco ligando v_i a v_j.

Antecessor Caso ocorra a uma relação inversa, v_j é antecessor de v_i.

Percursos (Boaventura)

Percurso (ou passeio)

É uma sequência (finita) de ligações sucessivas adjacentes (i.e. vértices e arestas $v_0, a_1, v_1, a_2, \ldots v_{k-1}, a_k, v_k$, onde $1 \leq i \leq k$) cada uma tendo uma extremidade adjacente à anterior e a outra subsequente (à exceção da primeira e da última).

Percursos (Boaventura)

Percurso (ou passeio)

É uma sequência (finita) de ligações sucessivas adjacentes (i.e. vértices e arestas $v_0, a_1, v_1, a_2, \dots v_{k-1}, a_k, v_k$, onde $1 \leq i \leq k$) cada uma tendo uma extremidade adjacente à anterior e a outra subsequente (à exceção da primeira e da última).

Exemplo:

Seja um grafo não vazio $G = (V, E)$, onde $V = \{1, 2, 3, 4, 5\}$ e $E = \{(1, 2), (1, 4), (2, 3), (2, 5), (3, 4), (4, 5), (5, 3)\}$.

$\rightarrow \{(1, 2), (2, 5), (5, 3), (3, 4)\}$ forma um percurso.

Percursos (Boaventura)

Percurso (ou passeio)

É uma sequência (finita) de ligações sucessivas adjacentes (i.e. vértices e arestas $v_0, a_1, v_1, a_2, \ldots v_{k-1}, a_k, v_k$, onde $1 \leq i \leq k$) cada uma tendo uma extremidade adjacente à anterior e a outra subsequente (à exceção da primeira e da última).

Exemplo:

Seja um grafo não vazio $G = (V, E)$, onde $V = \{1, 2, 3, 4, 5\}$ e $E = \{(1, 2), (1, 4), (2, 3), (2, 5), (3, 4), (4, 5), (5, 3)\}$.

$\rightarrow \{(1, 2), (2, 5), (5, 3), (3, 4)\}$ forma um percurso.

Obs.:

- É possível repetir vértices e arestas;
- Grafos orientados $\rightarrow$ considera-se apenas a adjacência sucessiva;
- Grafos não orientados $\rightarrow$ despreza-se a orientação das ligações.

Percursos (Boaventura)

Percurso fechado

Seja $G = (V, E)$ um grafo não vazio, um percurso fechado é um percurso em que a última ligação da sucessão é adjacente a primeira.

Percursos (Boaventura)

Percurso fechado

Seja $G = (V, E)$ um grafo não vazio, um percurso fechado é um percurso em que a última ligação da sucessão é adjacente a primeira.

Exemplo:

Seja um grafo não vazio $P = (V, E)$ da forma: $V = \{1, 2, 3, 4, 5\}$ e $E = \{(1, 2), (1, 4), (2, 3), (2, 5), (3, 1), (3, 4), (4, 5), (5, 3)\}$.

$\rightarrow \{(1, 2), (2, 3), (3, 1)\}$ forma um percurso fechado,
i.e. $v_0 = v_k = 1$

Percursos (Boaventura)

Percurso aberto

Seja $G = (V, E)$ um grafo não vazio, um percurso aberto é um percurso em que a última ligação da sucessão não é adjacente a primeira.

Percursos (Boaventura)

Percurso aberto

Seja $G = (V, E)$ um grafo não vazio, um percurso aberto é um percurso em que a última ligação da sucessão não é adjacente a primeira.

Exemplo:

Seja um grafo não vazio $P = (V, E)$ da forma: $V = \{1, 2, 3, 4, 5\}$ e $E = \{(1, 2), (1, 4), (2, 3), (2, 5), (3, 4), (4, 5), (5, 3)\}$.

$\rightarrow \{(1, 2), (2, 5), (5, 3), (3, 4)\}$ forma um percurso aberto, i.e. $v_0 = 1$ e $v_k = 4$.

Percursos (Boaventura)

Percurso Simples

Seja $G = (V, E)$ um grafo não vazio, um percurso simples é um percurso que não repete ligações (arestas).

Percursos (Boaventura)

Percurso Simples

Seja $G = (V, E)$ um grafo não vazio, um percurso simples é um percurso que não repete ligações (arestas).

Exemplo:

Seja um grafo não vazio $P = (V, E)$ da forma: $V = \{1, 2, 3, 4, 5\}$ e $E = \{(1,2), (1,4), (2,3), (2,5), (3,4), (4,5), (5,3)\}$.

$\rightarrow \{(1,2), (2,5), (5,3), (3,4)\}$ forma um percurso.

Percursos (Boaventura)

Percurso Elementar

Seja $G = (V, E)$ um grafo não vazio, um percurso elementar é um percurso que não repete vértices.

Obs.: Todo percurso elementar é simples, mas não o contrário.

Percursos (Boaventura)

Ciclo

Seja $G = (V, E)$ um grafo não vazio, um ciclo é um percurso simples e fechado. Ou seja, em um ciclo não há repetição de ligações (arestas).

Percursos (Boaventura)

Ciclo

Seja $G = (V, E)$ um grafo não vazio, um ciclo é um percurso simples e fechado. Ou seja, em um ciclo não há repetição de ligações (arestas).

Ciclo Simples

Seja $G = (V, E)$ um grafo não vazio, um ciclo simples é um ciclo que não repete vértices (exceto o primeiro).

Percursos (Boaventura)

Percurso Abrangente (spanning)

Seja um grafo $G = (V, E)$, um percurso abrangente é um percurso que considera todos os elementos de V ou de E pelo menos uma vez.

Cadeia ou Trilha (Goldbarg)

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

Cadeia (ou trilha)

Uma cadeia ou trilha é um passeio (percurso)
sem repetição de arestas.

Cadeia = percurso simples

Obs.: Os conceitos de aberto e fechado são aplicados também a cadeia e caminho.

Comprimento de um percurso

Comprimento de um percurso

O comprimento de um percurso é dado pelo número de ligações que ele utiliza, contando-se as repetições.

Exemplo:

Seja um grafo não vazio $P = (V, A)$ da forma: $V = \{1, 2, 3, 4, 5\}$ e $A = \{(1, 2), (1, 4), (2, 3), (2, 5), (3, 4), (4, 5), (5, 3)\}$.
$\{(1, 2), (2, 5), (5, 3), (3, 4)\}$ forma um percurso de tamanho 4.

Caminho

Caminho [Diestel2010]

Um caminho é um grafo não vazio $P = (V, A)$ da forma:
$V = \{x_0, x_1, \ldots, x_k\}$ e $A = \{x_0x_1, x_1x_2, \ldots, x_{k-1}x_k\}$ onde todos os (vértices) x_i são distintos.

x_0 e x_k são os <u>vértices finais</u>. x_1 e x_{k-1} são os <u>vértices interiores</u>.

Cadeia sem repetição de vértices = caminho (Goldbarg)

Caminho

Caminho [Diestel2010]

Um caminho é um grafo não vazio $P = (V, A)$ da forma: $V = \{x_0, x_1, \ldots, x_k\}$ e $A = \{x_0x_1, x_1x_2, \ldots, x_{k-1}x_k\}$ onde todos os (vértices) x_i são distintos.

x_0 e x_k são os vértices finais. x_1 e x_{k-1} são os vértices interiores.

Cadeia sem repetição de vértices = caminho (Goldbarg)

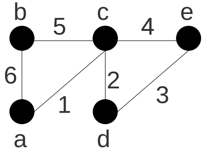
Caminho

Caminho [Diestel2010]

Um caminho é um grafo não vazio $P = (V, A)$ da forma:
$V = \{x_0, x_1, \ldots, x_k\}$ e $A = \{x_0x_1, x_1x_2, \ldots, x_{k-1}x_k\}$ onde todos os (vértices) x_i são distintos.

x_0 e x_k são os <u>vértices finais</u>. x_1 e x_{k-1} são os <u>vértices interiores</u>.

Cadeia sem repetição de vértices = caminho (Goldbarg)



Percurso Fechado de Euler

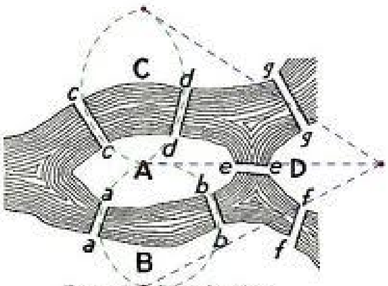


FIGURE 98. Geographic Map:
The Königsberg Bridges.

Percurso Fechado de Euler

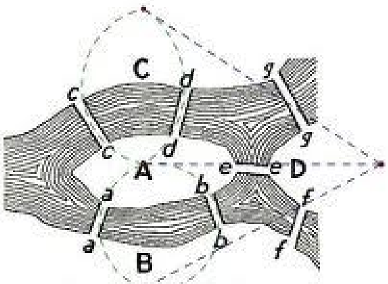


Figure 98.* Geographic Map: The Königsberg Bridges.

O problema das sete pontes de Königsberg

A partir de qualquer ponto da cidade é possível caminhar atravessando cada ponte uma única vez e retornar ao ponto inicial?

Percurso Fechado de Euler

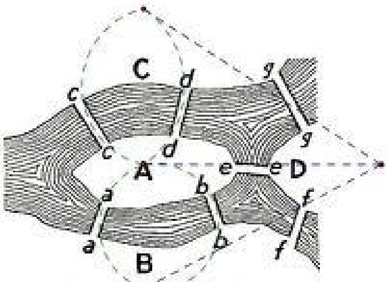


FIGURE 98. Geographic Map: The Königsberg Bridges.

O problema das sete pontes de Königsberg

A partir de qualquer ponto da cidade é possível caminhar atravessando cada ponte uma única vez e retornar ao ponto inicial?

pontes = arestas; vértices = regiões delimitadas pelo rio

Percursos Eulerianos

Percurso Euleriano

Seja $G = (V, E)$, um percurso aberto ou fechado é euleriano se ele utiliza cada ligação do grafo uma única vez.

Percursos Eulerianos

Percurso Euleriano

Seja $G = (V, E)$, um percurso aberto ou fechado é euleriano se ele utiliza cada ligação do grafo uma única vez.

Aplicações

Problemas de atendimento sequencial. Ex: entrega de correio, coleta de lixo, vendas, ...

Percursos Eulerianos

Grafo Euleriano (Goldbarg)

É um grafo que possui uma cadeia fechada (i.e. sem repetição de arestas) que passa por todas as arestas.

Percursos Eulerianos

Grafo Euleriano (Goldbarg)

É um grafo que possui uma cadeia fechada (i.e. sem repetição de arestas) que passa por todas as arestas.

Grafo semieuleriano (Goldbarg)

É um grafo que possui uma cadeia (i.e. sem repetição de arestas) que passa por todas as arestas.

Percursos Eulerianos

Grafo Euleriano (Goldbarg)

É um grafo que possui uma cadeia fechada (i.e. sem repetição de arestas) que passa por todas as arestas.

Grafo semieuleriano (Goldbarg)

É um grafo que possui uma cadeia (i.e. sem repetição de arestas) que passa por todas as arestas.

Teorema de Euler (Boaventura)

Um grafo $G = (V, A)$ conexo e não orientado possui um ciclo euleriano (cadeia fechada) se e somente se não possuir vértices de grau ímpar.

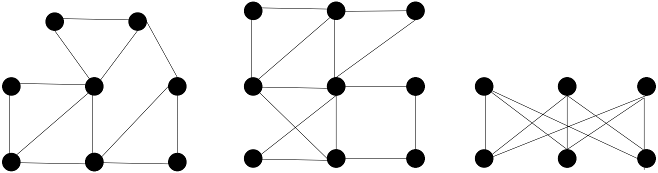
Percursos Eulerianos Abertos (Grafo semieuleriano)

Teorema (Gross):

Um grafo conectado G possui um percurso euleriano aberto (cadeia) se e somente se ele possui exatamente dois vértices de grau ímpar. Além disso, os vértices iniciais e finais da cadeia euleriana precisam ser os dois vértices de grau ímpar.

Exemplo

Verifique se os seguinte grafos são Eulerianos ou se possuem percursos aberto de Euler



O algoritmo de Fleury

Seja um grafo $G = (V, E)$, o algoritmo de Fleury a seguir obtém um *percurso euleriano fechado* a partir de um vértice qualquer.

Entrada: Grafo conexo $G = (V, E)$
Saída: um percurso fechado de Euler C
Escolher um vertice v qualquer de G; $C \leftarrow \{v\}$
repita
  Escolher uma aresta (v, w) nao marcada via regra da ponte
  atravessar (v,w); $C \leftarrow C\{w\}$
  Marcar (v,w); $v \leftarrow w$
até *todas as arestas estejam marcadas*;
$C \leftarrow C\{v\}$; Imprimir C

Regra da Ponte: Se uma aresta (v, w) é uma ponte no grafo induzido pelas arestas não marcadas, então (v, w) só deve ser escolhida pelo algoritmo caso não exista qualquer outra opção.

O algoritmo de Fleury

Para m arestas.

- A cada iteração uma aresta é marcada $\rightarrow$ m iterações no laço;
- A cada iteração teste de condição de ponte $\rightarrow$ pode ser realizado em $O(m)$ através de em algoritmo de percurso.

O algoritmo de Fleury

Para m arestas.

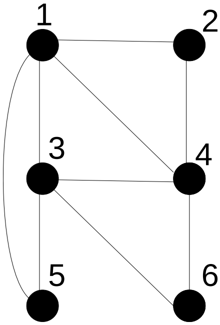
- A cada iteração uma aresta é marcada $\rightarrow$ m iterações no laço;
- A cada iteração teste de condição de ponte $\rightarrow$ pode ser realizado em $O(m)$ através de em algoritmo de percurso.

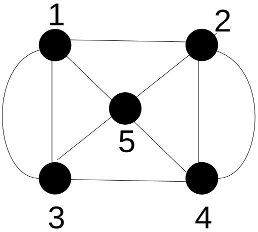
Complexidade (sem otimização)

$O(m^2)$

Exemplo

Utilizando o algoritmo de Fleury, encontre um percurso euleriano fechado nos seguintes grafo:





Percursos Hamiltonianos

Percurso Hamiltoniano

Seja $G = (V, E)$, um percurso aberto ou fechado é hamiltoniano se ele utiliza cada vértice do grafo uma única vez.

Percursos Hamiltonianos

Percurso Hamiltoniano

Seja $G = (V, E)$, um percurso aberto ou fechado é hamiltoniano se ele utiliza cada vértice do grafo uma única vez.

Aplicações

Problemas relacionados a atendimentos sequenciais em locais relativamente distantes uns dos outros ou em pontos caracterizados por sua posição e não pela sequência de atendimento.

O segundo caso resulta na procura de elementos no conjunto das pemutações nos vértices do grafo.

Percursos Hamiltonianos

Grafo hamiltoniano

É um grafo que possui um ciclo hamiltoniano.

Percursos Hamiltonianos

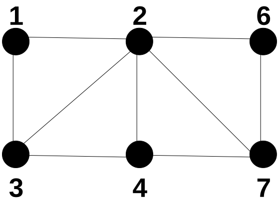
Grafo hamiltoniano

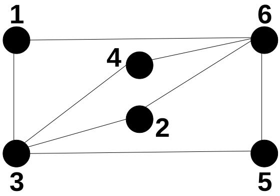
É um grafo que possui um ciclo hamiltoniano.

- Não existe uma forma eficiente para determinar se um dado grafo é hamiltoniano.
- A verificação se um grafo é hamiltoniano consiste em um problema NP-Completo.

Exercício

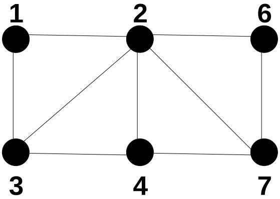
Verifique se os seguinte grafos são Eulerianos/Hamiltonianos

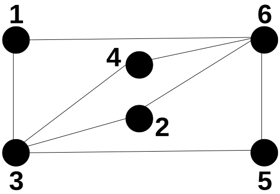




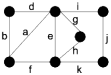
Exercício

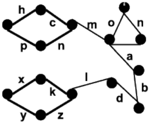
Verifique se os seguinte grafos são Eulerianos/Hamiltonianos





Verifique se os seguintes grafos contém um percurso aberto ou fechado Eulerianos, e apresente o percurso caso exista





Percursos Hamiltonianos

O Problema do Caixeiro Viajante (PCV)

Seja um grafo G tal que cada uma de suas arestas possuem um peso, o problema do caixeiro viajante consiste em procurar um ciclo hamiltoniano em G. minimizando a soma dos pesos das arestas que o compõem.

Percursos Hamiltonianos

O Problema do Caixeiro Viajante (PCV)

Seja um grafo G tal que cada uma de suas arestas possuem um peso, o problema do caixeiro viajante consiste em procurar um ciclo hamiltoniano em G. minimizando a soma dos pesos das arestas que o compõem.

Ou seja, o problema do Caixeiro Viajante consiste em determinar um percurso hamiltoniano fechado de valor mínimo.

Percursos Hamiltonianos

O Problema do Caixeiro Viajante (PCV)

Seja um grafo G tal que cada uma de suas arestas possuem um peso, o problema do caixeiro viajante consiste em procurar um ciclo hamiltoniano em G. minimizando a soma dos pesos das arestas que o compõem.

Ou seja, o problema do Caixeiro Viajante consiste em determinar um percurso hamiltoniano fechado de valor mínimo.

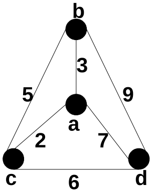
Exemplo

Supondo que o caixeiro viajante precisa visitar várias cidades no próximo mês. O peso das arestas nos sequintes grafos representam o custo de viagens entre cada par de cidades. Encontre um ciclo hamiltoniano cuja soma total das arestas é mínima.

Obs.: Problema NP-Difícil

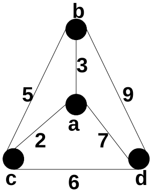
Exemplo Caixeiro Viajante

Qual o percurso hamiltoniano (circuito) de tamanho mínimo, partindo do vético a*, no seguinte grafo:*



Exemplo Caixeiro Viajante

Qual o percurso hamiltoniano (circuito) de tamanho mínimo, partindo do vétice a*, no seguinte grafo:*



$|abcda| = 3 + 5 + 6 + 7 = 21$
$|acbda| = 2 + 5 + 9 + 7 = 23$
$|acdba| = 2 + 6 + 9 + 3 = 20$

Um Algoritmo Heurístico para o PCV

Algoritmo de Bellmore e Nemhauser

Entrada: Grafo $G = (V, E)$ com n vertices

Saída: Um ciclo hamiltoniano (possivelmente) de custo mínimo

Escolher um vertice inicial v_i qualquer; $H \leftarrow \{v_i\}$

enquanto $|H| < n$ **faça**

  Encontrar o vertice $v_k \notin H$ mais proximo de v_i

  $H \leftarrow H\{v_k\}$

  $i \leftarrow k$

fim

Um Algoritmo Heurístico para o PCV

Algoritmo de Bellmore e Nemhauser

Entrada: Grafo $G = (V, E)$ com n vertices
Saída: Um ciclo hamiltoniano (possivelmente) de custo mínimo
Escolher um vertice inicial v_i qualquer; $H \leftarrow \{v_i\}$
enquanto $|H| < n$ **faça**
    Encontrar o vertice $v_k \notin H$ mais proximo de v_i
    $H \leftarrow H\{v_k\}$
    $i \leftarrow k$
fim

Complexidade:

- No laço percorrer n vértices;
- Para cada vértice, examinar todos os vértices vizinhos (max. $O(n)$ vizinhos);
- Complexidade: $O(n^2)$

Bibliografia

- NETTO, Paulo O. B. **Teoria e Modelos e Algoritmos**, 4ª. ed. Edgard Blücher. São Paulo, 2006;
- GOLDBARG, Marco, GOLDBARG, Elizabeth, **Grafos: Conceitos, algoritmos e aplicações**, 1a. ed. Elsevier, 2012.