INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
SUL-RIO-GRANDENSE
Campus Passo Fundo

Ciência da Computação

ESTRUTURA DE DADOS II

Prof. Adilso Nunes de Souza

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA SUL-RIO-GRANDENSE Campus Passo Fundo

MENOR CAMINHO

- O menor caminho entre dois vértices é a aresta que os conecta. No entanto, é muito comum em um grafo não existir uma aresta conectando dois vértices, isto é, eles não são adjacentes.
- Apesar disso, dois vértices podem ser conectados por uma sequência de arestas.
- Caso essa seja a menor sequência de arestas dizemos que é o **menor caminho**, **caminho mais curto** ou **caminho geodésico** entre eles.

MENOR CAMINHO

- Segundo BACKES, 2016 o menor caminho entre dois vértices é o caminho que apresenta o menor comprimento dentre todos os possíveis que conectam esses vértices.
- Geralmente o comprimento se refere ao número de arestas que conectam os dois vértices.
- Considerando um grafo ponderado, podemos calcular o menor caminho com a soma dos pesos das arestas que compõem o caminho.

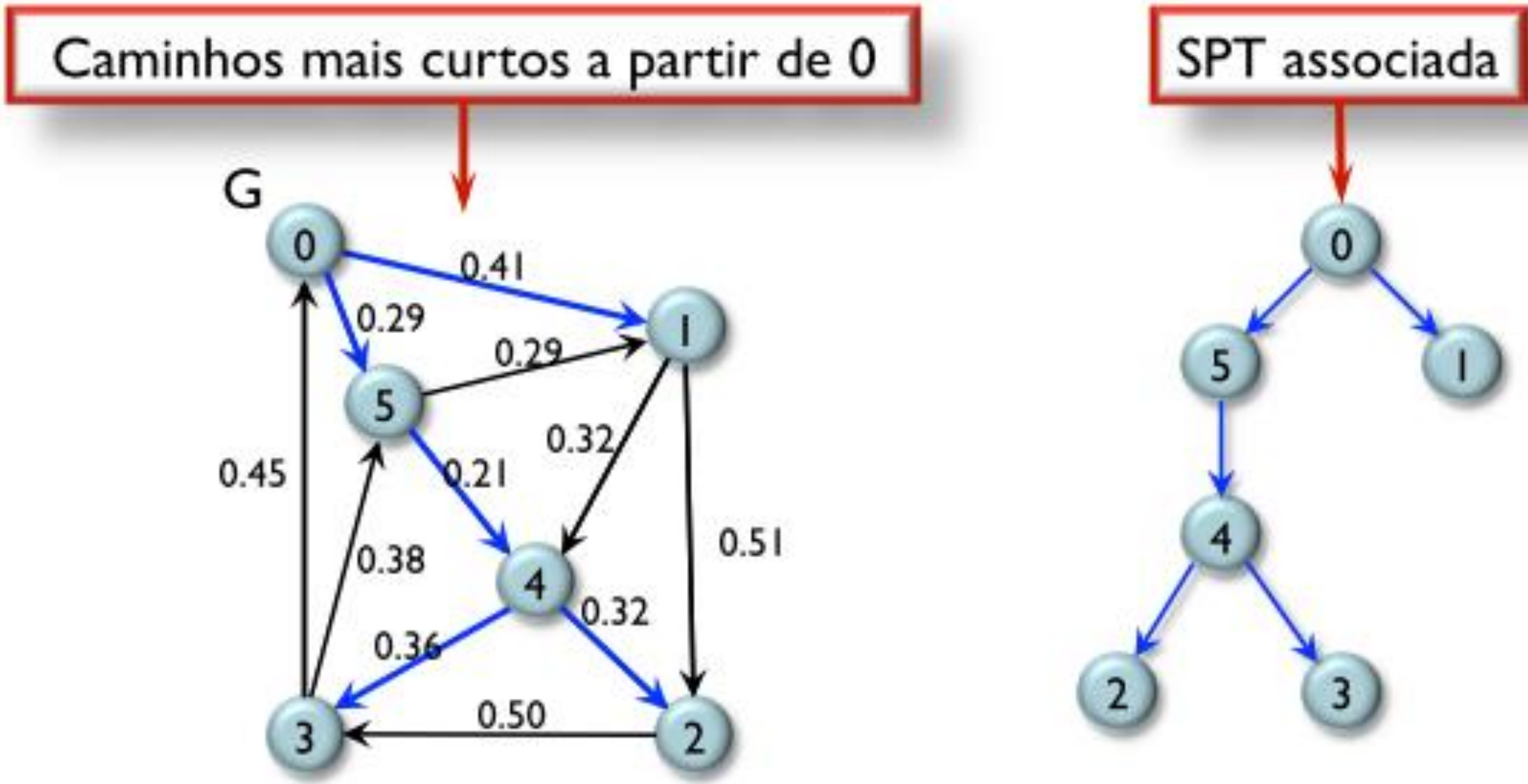
MENOR CAMINHO

- Em grafos não ponderados (ou arestas de mesmo peso) a busca em largura soluciona o problema do caminho mais curto entre dois vértices.
- São conhecidos como problemas de caminhos mais curtos - “shortest path problems”.
- Para estes algoritmos é necessário utilizar uma estrutura do tipo árvore, denominada SPT (Shortest Path Tree) árvore de caminho mais curto.

SPT

- É um subgrafo contendo s e todos os vértices alcançáveis a partir de s que forma uma árvore direcionada com raiz em s tal que todo caminho da árvore é um caminho mínimo no grafo.

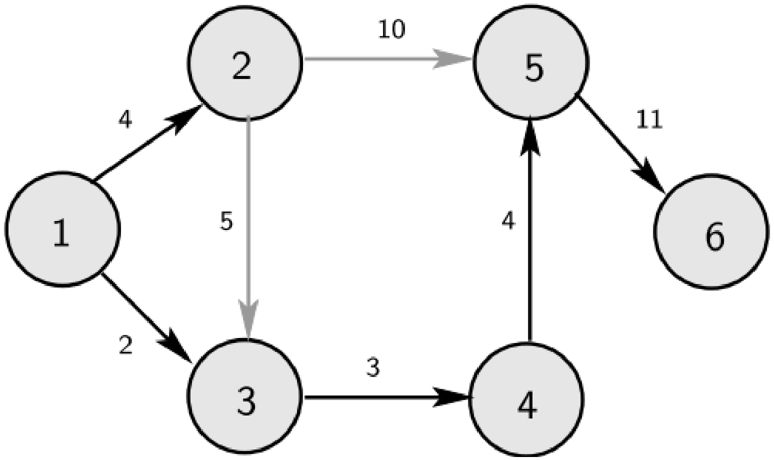
SPT


Caminhos mais curtos a partir de 0
SPT associada
G
0
1
2
3
4
5
0.41
0.29
0.29
0.32
0.21
0.45
0.51
0.38
0.32
0.36
0.50

SPT

- A árvore de caminhos mínimos (SPT) pode ser representada por dois vetores indexados pelos vértices v
  - dist[v]: armazena o comprimento do caminho mínimo entre s e v.
  - prev[v]: armazena a última aresta do caminho mínimo entre s e v.

SPT

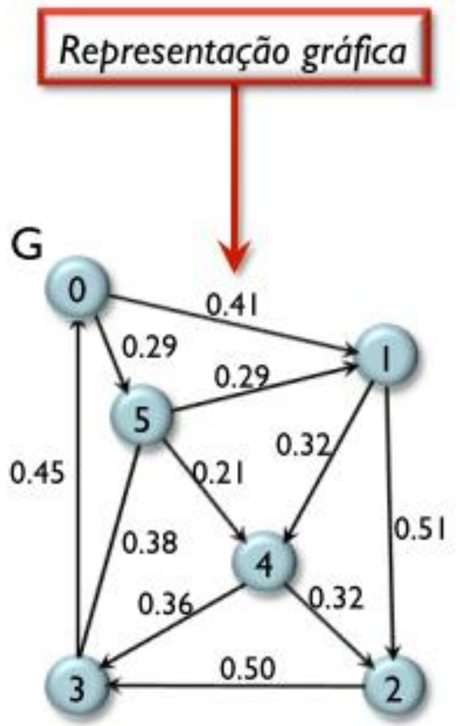


v	prev[v]	dist[v]
1	–	0
2	(1,2)	4
3	(1,3)	2
4	(3,4)	5
5	(4,5)	9
6	(5,6)	20

- OBS: Árvore de caminhos mínimos representado pelas setas em preto.

TIPOS DE BUSCA

- Problema 1 - Caminho mais curto fonte-destino:
  - Dado um vértice inicial s e um vértice destino t, qual o caminho mais curto no grafo de s para t



S = 0

T = 2

Caminhos possíveis:

0 – 1 – 2: 0.92

0 – 1 – 4 – 2: 1.05

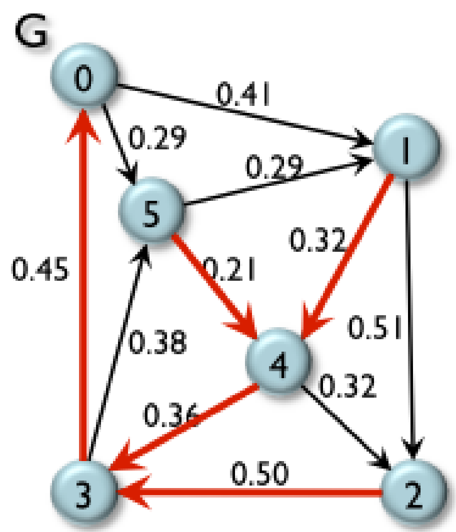
0 – 5 – 4 – 2: 0.82

Caminho mais curto:

0 – 5 – 4 – 2: 0.82

TIPOS DE BUSCA

- Problema 2 - Caminhos mais curtos de fonte única:
  - Dado um vértice inicial s, quais os caminhos mais curtos que ligam todos os outros vértices à s?
  - S = 0

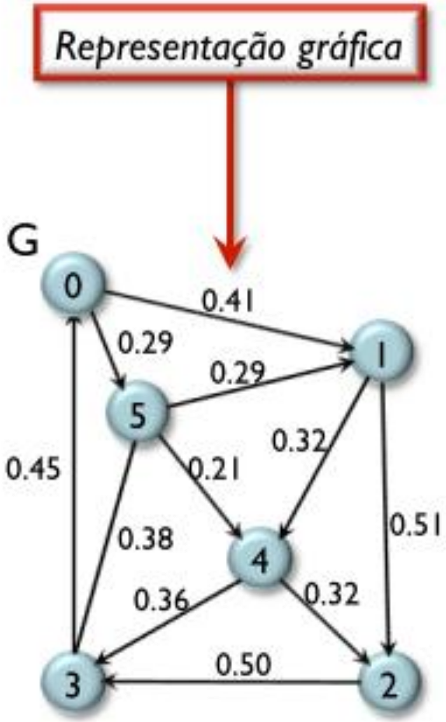


	0
1.13	1-4-3-0
0.95	2-3-0
0.45	3-0
0.81	4-3-0
1.02	5-4-3-0

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
SUL-RIO-GRANDENSE
Campus Passo Fundo

TIPOS DE BUSCA

- Encontre os caminhos mais curtos para os diferentes casos:
  - S = 1
  - S = 4
  - S = 5



TIPOS DE BUSCA

- Problema 3 - Caminhos mais curtos entre todos:
  - Quais os caminhos mais curtos ligando todos os vértices de um grafo?
  - Pode ser resolvido aplicando o algoritmo N vezes, uma vez para cada vértice origem.

APLICAÇÕES

- Rotas de veículos
- Planejamento de tráfego urbano
- Navegação robótica
- Roteamento em telecomunicações
- Mapa de conexões de voo
- Redes elétricas e telefônicas

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA SUL-RIO-GRANDENSE Campus Passo Fundo

PESOS DAS ARESTAS

- Intuitivamente as arestas denotam distâncias, mas pesos de arestas podem representar outras medidas que não sejam distâncias, como tempo, custo, multas, prejuízos ou qualquer outra quantidade que se acumule linearmente ao longo de um caminho e que seria interessante minimizar.

ALGORITMOS

- Exemplos de algoritmos para calcular o menor caminho:
  - Prim
  - Kruskal
  - Bellman-Ford
  - Dijkstra
  - Floyd-Warshall

PRIM

- O algoritmo de PRIM é um algoritmo clássico capaz de obter uma solução ótima para o problema da árvore geradora mínima.
- Uma árvore geradora mínima (spanning tree) é um subgrafo que contém todos os vértices do grafo original e um conjunto de arestas que permite conectar todos esses vértices na forma de uma árvore.

PRIM

- Para um grafo possuir uma árvore geradora mínima ele deve satisfazer as seguintes propriedades: não direcionado, conexo e ponderado.
- O algoritmo de PRIM inicia a árvore com um vértice qualquer e adiciona um novo vértice à árvore a cada iteração. Esse processo continua até que todos os vértices do grafo façam parte da árvore, ou não seja possível achar uma aresta de menor peso conectando os vértices.

PRIM

- O algoritmo de PRIM pode ser implementado com uma função que recebe o grafo, o vértice que será o ponto de partida para crescer a árvore e um array para marcar quem é o pai de cada vértice na árvore resultante.

KRUSKAL

- O algoritmo de KRUSKAL é outro algoritmo clássico capaz de obter uma solução ótima para o problema da árvore geradora mínima.
- Considerando cada vértice uma árvore independente, o algoritmo procura a aresta de menor peso que conecta duas árvores diferentes. Os vértices das árvores selecionadas passam a fazer parte de uma mesma árvore.
- O processo se repete até que todos os vértices façam parte de uma mesma árvore.

KRUSKAL

- O algoritmo de KRUSKAL se inicia com uma floresta, várias árvores. A cada iteração, duas árvores são selecionadas para ser unidas em uma mesma árvore.
- Esse algoritmo pode ser implementado com uma função que recebe três parâmetros: o grafo, o vértice que será ponto de partida e um array para marcar quem é o pai de cada vértice. Também será necessário um array auxiliar para gerenciar a qual árvore cada vértice pertence.

DIJKSTRA

- O algoritmo de Dijkstra determina a árvore de caminhos mínimos a partir de s para qualquer grafo ponderado com pesos não negativos nas arestas.
- Mantém um conjunto de S vértices cujos pesos finais de caminho mínimo que parte da fonte S já foram determinados.
- O tempo de execução do algoritmo Dijkstra é inferior ao algoritmo de Bellman-Ford.

DIJKSTRA

- Coloca-se o vértice fonte na raiz da árvore e constrói-se a árvore uma aresta de cada vez.
- Toma-se sempre a aresta que estabelece o caminho mais curto do vértice fonte a algum vértice que ainda não esteja na SPT.
- Adicionam-se vértices à SPT por ordem crescente da sua distância (através da SPT) ao vértice de partida.

DIJKSTRA

- Este algoritmo pode ser visto como um método de procura generalizada, diferente da DFS, da BFS e do algoritmo de Prim apenas pela regra de entrada de arestas na árvore.

BELLMAN-FORD

- Publicado em 1956 por Ford, em 1958 por Bellman e em 1957 por Edward F. Moore.
- Também conhecido como algoritmo de Bellman-Ford-Moore.
- Menos eficiente do que Dijkstra, mas trata arestas com pesos negativos. É capaz de detectar ciclos negativos.

BELLMAN-FORD

- O algoritmo de Bellman-Ford resolve o problema de caminhos mínimos de fonte única no caso geral no qual os pesos das arestas podem ser negativos.
- O algoritmo devolve um valor booleano que indica se existe ou não um ciclo de peso negativo que pode ser alcançado da fonte.
- Se tal ciclo existe, o algoritmo indica que não há nenhuma solução. Caso contrário produz os caminhos mínimos e seus pesos.

FLOYD-WARSHALL

- O algoritmo de Floyd-Warshall é um algoritmo que resolve o problema de calcular o caminho mais curto entre todos os pares de vértices em um grafo orientado e valorado.
- O algoritmo Floyd-Warshall foi publicado por Robert Floyd em 1962.
- Conhecido como: Floyd’s Algorithm, Roy-Warshall Algorithm, Roy-Floyd Algorithm.

FLOYD-WARSHALL

- O algoritmo de Floyd-Warshall considera os vértices intermediários de um caminho mínimo, onde um vértice intermediário de um caminho simples p = (v1, v2, ..., vi) é qualquer vértice de p exceto v1 ou vi, isto é qualquer vértice no conjunto(v2, v3,..., vi-1)
- Existe uma variedade de métodos diferentes para construir caminhos mínimos no algoritmo de Floyd-Warshall

REFERÊNCIAS

- PEREIRA, Silvio do Lago. Estrutura de Dados Fundamentais: Conceitos e Aplicações, 12. Ed. São Paulo, Érica, 2008.
- BACKES, André Ricardo, Estrutura de dados descomplicada: em linguagem C, 1 Ed. – Rio de Janeiro: Elsevier, 2016.
- ROCHA, Anderson, Grafos – Representações, buscas e Aplicações.
- SENGER, H., Notas de Aula, Universidade de São Judas Tadeu, 1999.
- WALDEMAR Celes, Renato Cerqueira, José Lucas Rangel, Introdução a Estruturas de Dados, Editora Campus (2004).
- VELOSO, Paulo. SANTOS, Celso dos. AZEVEDO, Paulo. FURTADO, Antonio. Estrutura de dados. Rio de Janeiro: Ed. Elsevier, 1983 27ª reimpressão.
- https://www.cs.usfca.edu/~galles/visualization/Algorithms.html
- ZIVIANI, Nivio. Projeto de Algoritmos com implementações em Java e C++, 2007.