Grafos
representação e aplicações

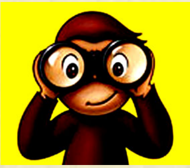
Prof. Guilherme Tomaschewski Netto
guilherme.netto@gmail.com

Roteiro

- Contextualização
- Apresentação, um pouco de história
- Conceitos Grafos
- Principais aplicacões
- Estruturas de Dados para Grafos
- Algoritmos de Busca
- Tipos de grafos e outras aplicações

Legendas

- Nesta apresentação serão utilizadas algumas legendas:



Indica uma referência, para quem ficou curioso e quer aprofundar mais seus conhecimentos sobre o assunto

Indica uma referência importante, leitura obrigatória.

Competências desejadas

Para compreênsão dos conceitos abordados é desejado que os alunos já tenham apropriado as seguintes competências:

- Capacidade de implementar algoritmos básicos
- Conhecimento sobre tipos de dados
- Entendimento sobre matrizes e listas lineares

Histórico

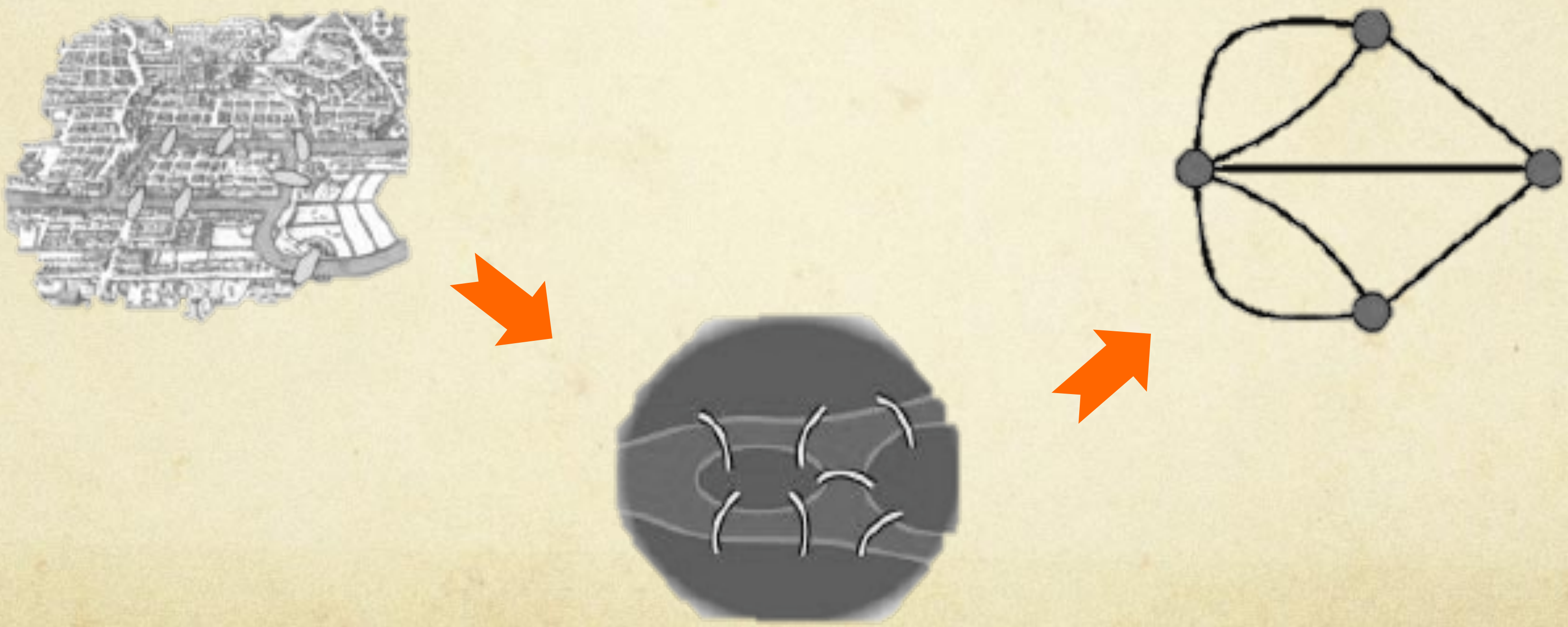
- A teoria de grafos tem uma origem relativamente recente (século XVIII) na história da matemática.
- O primeiros cientistas : L. Euler, G. Kirchhoff e A. Cayley.
- A teoria de grafos tem extensiva utilização em matemática aplicada, pois demonstra ser uma poderosa ferramenta para a modelagem de diversas situações reais em, entre outros, física, química, biologia, engenharia e pesquisa operacional.

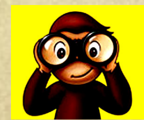
Pontes de Konigsberg

- O primeiro e mais famoso problema em teoria de grafos foi resolvido por Euler em 1736. Na cidade de Konigsberg sete pontes cruzam o rio Pregel estabelecendo ligações entre duas ilhas e entre as ilhas e as margens opostas do rio.

Pontes de Konigsberg

Será possível fazer um passeio pela cidade, começando e terminando no mesmo lugar, cruzando cada ponte exatamente uma vez?

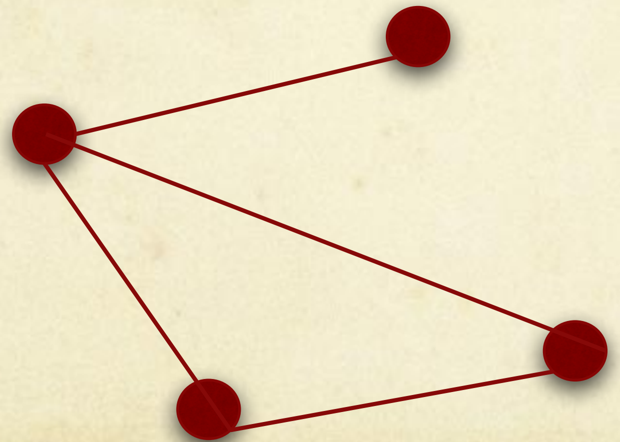




http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0101-74382009000200005

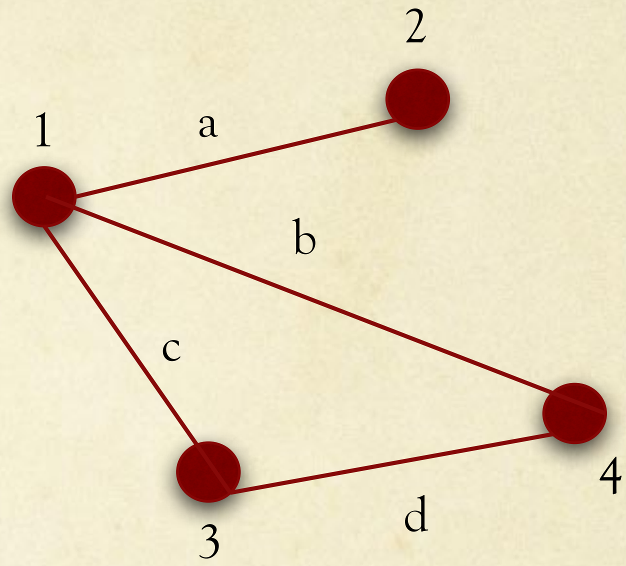
Grafo

Uma noção simples, abstrata e intuitiva, usada para representar a ideia de alguma relação entre os objetos. Graficamente, aparece representado por uma figura com nós ou vértices, significando os objetos, unidos por um traço denominado aresta ou arcos, configurando a relação imaginada.



Representação Matemática

G = (V,A)

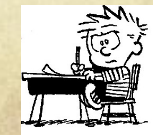


Onde:

- V é o conjunto de vértices
- A é o conjunto de arestas

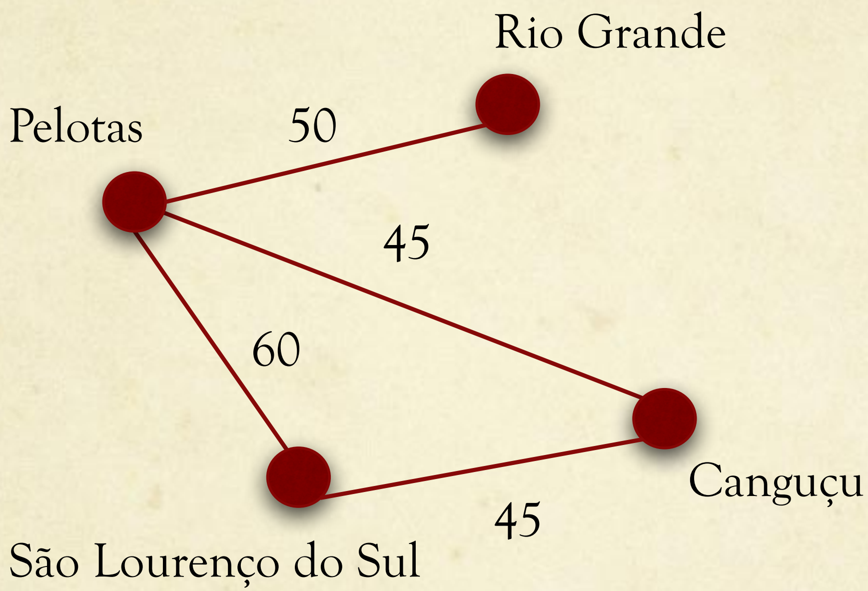
V={1,2,3,4}

A={{1,2},{1,3},{1,4},{3,4}}



ZIVIANI, 2002. Página 252-256

Exemplo



Para Pensar

Seja o grafo G(V,A) dado por:

- V = { p | p é uma pessoa }
- A = { (u,v) | < u é amigo de v > }

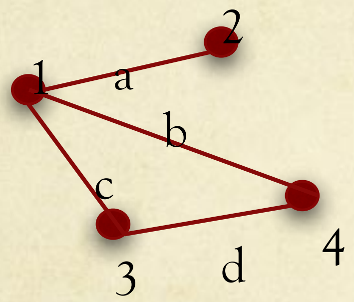
Seja o exemplo:

- V = { Maria, Pedro, Joana, Luiz }
- A = { (Maria, Pedro) , (Joana, Maria) , (Maria,
- Luiz), (Pedro, Luiz) , (Joana, Pedro) }

Construa o grafo do exemplo acima.

Tipos de Grafos

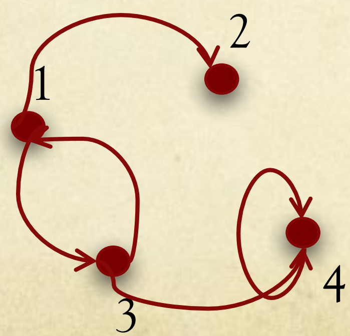
- Grafo não direcionado ou não orientado



V={1,2,3,4}

A={{1,2},{1,3},{1,4},{3,4}}

- Grafo direcionado ou orientado



V={1,2,3,4}

A={{1,2},{1,3},{3,1},{3,4},{4,4}}

Definições

- Vértices Adjacentes

  Um vértice *v* é adjacente ao vértice *u* se existir uma aresta que ligue *v* até *u*. Nos grafos não direcionados a relação de adjacência é simétrica, o que pode não ocorrer nos grafos direcionados.

- Grau de um vértice

  Para um grafo não direcionado é o número de arestas que incidem sobre ele. Em um grafo direcionado é a soma das arestas que chegam mais as que saem.

Definições

- O percurso entre dois vértices é denominado caminho;
- O comprimeto de um caminho é determinado pelo número de arestas a que este pertencem;
- Um grafo é considerado **conectado** se cada par de vértices está conectado por um caminho, ou seja, são conjuntos de vértices sob a relação “é alcançável a partir de”.

Aplicações

- Motores de busca de páginas Web
  - Vértices são as páginas HTML e as arestas (direcionadas) são links ligando as páginas
  - Identificar proximidade entre duas páginas quaisquer
  - Identificar se uma página é acessível a partir de outra
  - Identificar o número de links para uma página (grau do vértice)

Aplicações

- Problema de Fluxo em Redes

- Roteamento de Carga
  - Vértices são pontos de entrega e as arestas (com pesos) são estradas
  - Descobrir a rota de entrega com menor custo
  - Identificar a rota mais rápida

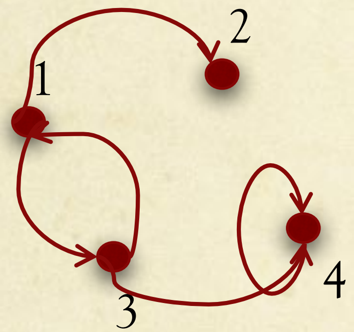
Aplicações

- Você poderia citar outras situações que poderiam ser modeladas através de Grafos?

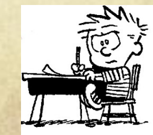
    Em grupos discutam e criem uma lista para expor à turma.

Matrizes de Adjacência

- É uma representação por meio de uma matriz quadrada das arestas de um determinado grafo.

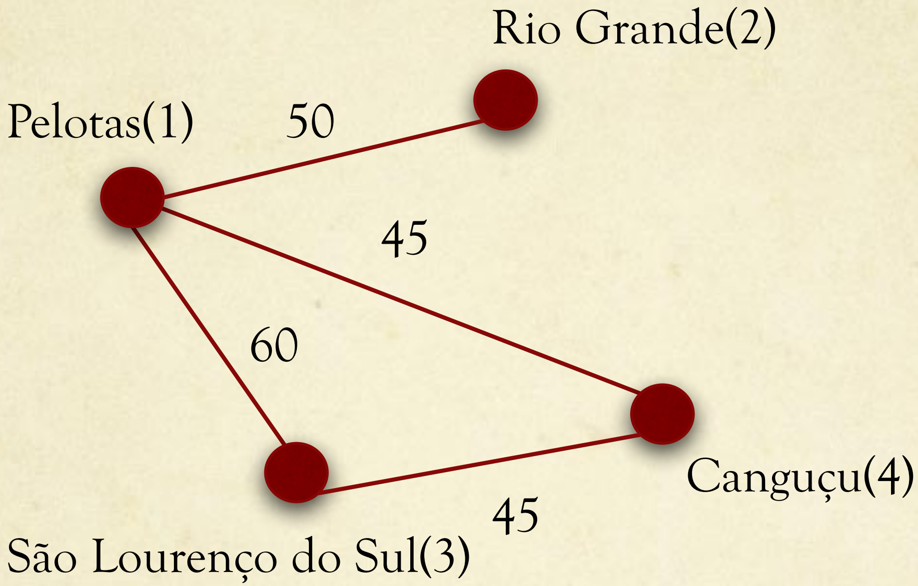


	1	2	3	4
1		1	1	
2				
3	1			1
4				1



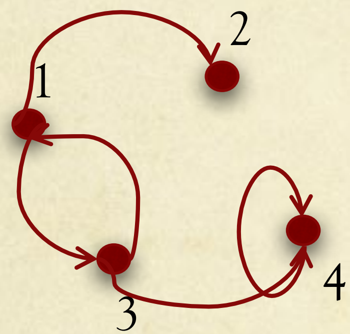
ZIVIANI, 2002. Página 257-260

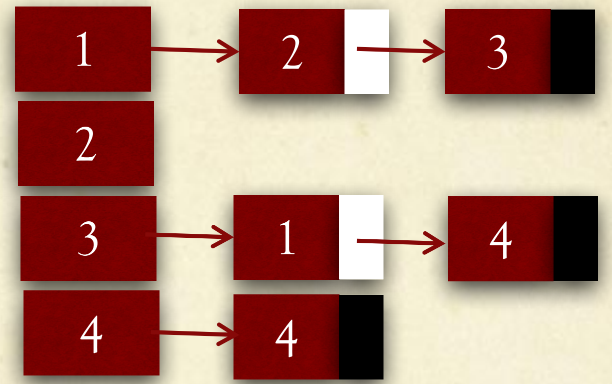
Matrizes de Adjacência

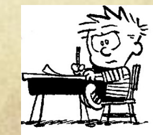


	1	2	3	4
1		50	60	45
2	50			
3	60			45
4	45		45	

Lista de Adjacência

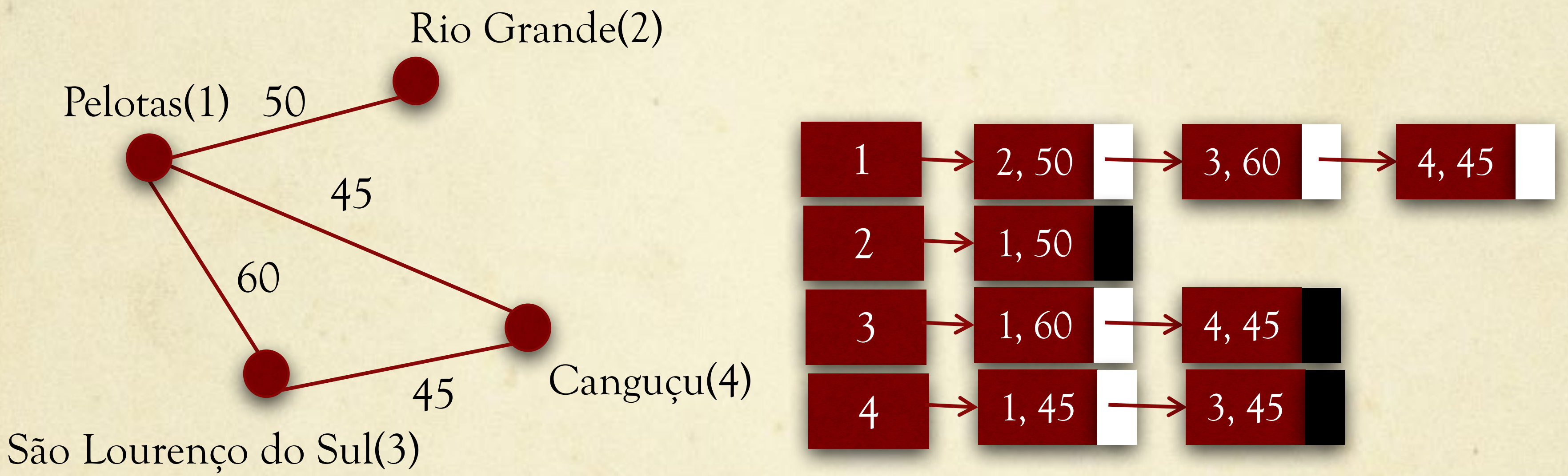






ZIVIANI, 2002. Página 261-264

Lista de Adjacência

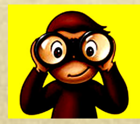


Lista X Matriz

- Matriz primeira representação
- Algorítmos mais rápidos
- Menor otimização de memória
- Grafos não-dirigidos possuem matrizes simétricas

Para Pensar

No grafo criado no exercício anterior, montar a Matriz e a Lista de Adjacência equivalentes.



http://gdias.visaodigital.pt/projecto/pt/info_represent.php

Algoritmos de Busca

Um caminhamento (Busca) é um procedimento sistemático para explorar um grafo examinando todos os seus vértices e arestas. Estudaremos dois algoritmos de caminhamento:

- Busca em Largura
  - Base para outros algoritmos, como Dijktra, que encontra caminhos mais curtos.
- Busca em Profundidade
  - Encontrar caminhos entre vértices

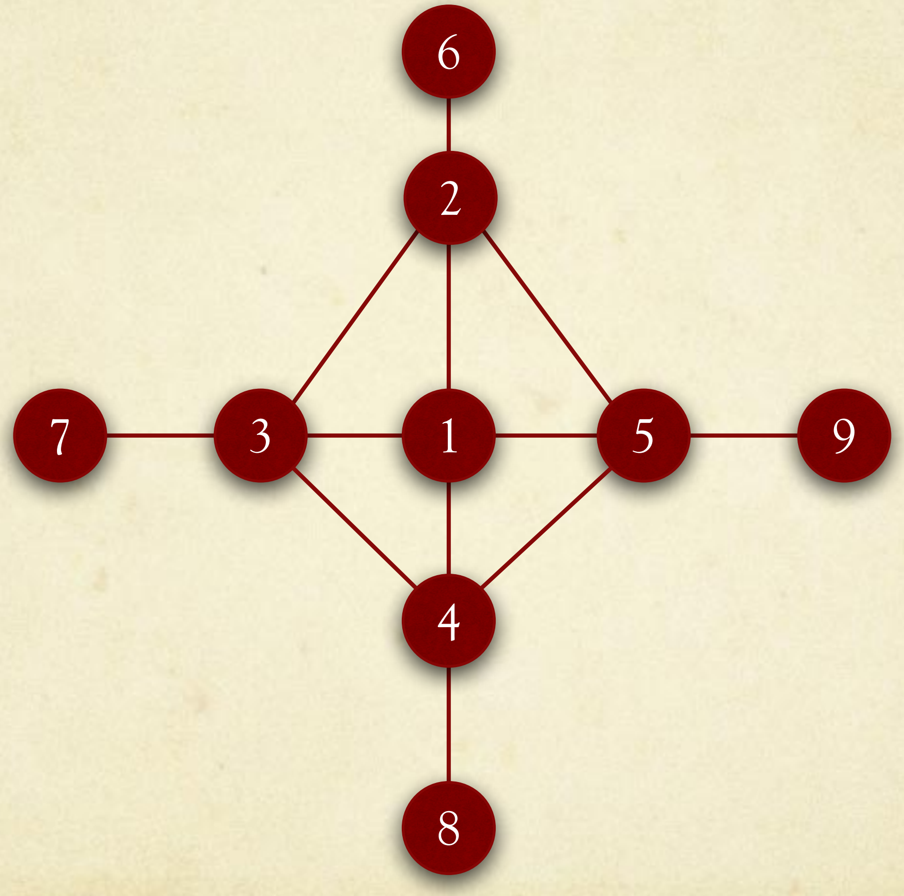
Busca em Largura (Breadth –First Search - BFS)

- A busca em largura expande a exploração de um grafo em níveis.
- A partir do vértice inicial, o nível explorado é o dos vértices adjacentes;
- Após a exploração deste nível, passa-se à exploração dos vértices adjacentes aos do nível anterior;
- Caso o grafo seja desconectado, ao fim da exploração de um componente passa-se ao próximo;
- O procedimento se repete até que todos os vértices tenham sido explorados.

BFS

- Uma estrutura de fila é utilizada para guiar os passos da busca;
- Durante a exploração, um vértice é descoberto na primeira vez em que é encontrado, quando é então enfileirado
- Quando o vértice é retirado da fila, ele é considerado visitado ou terminado

BFS



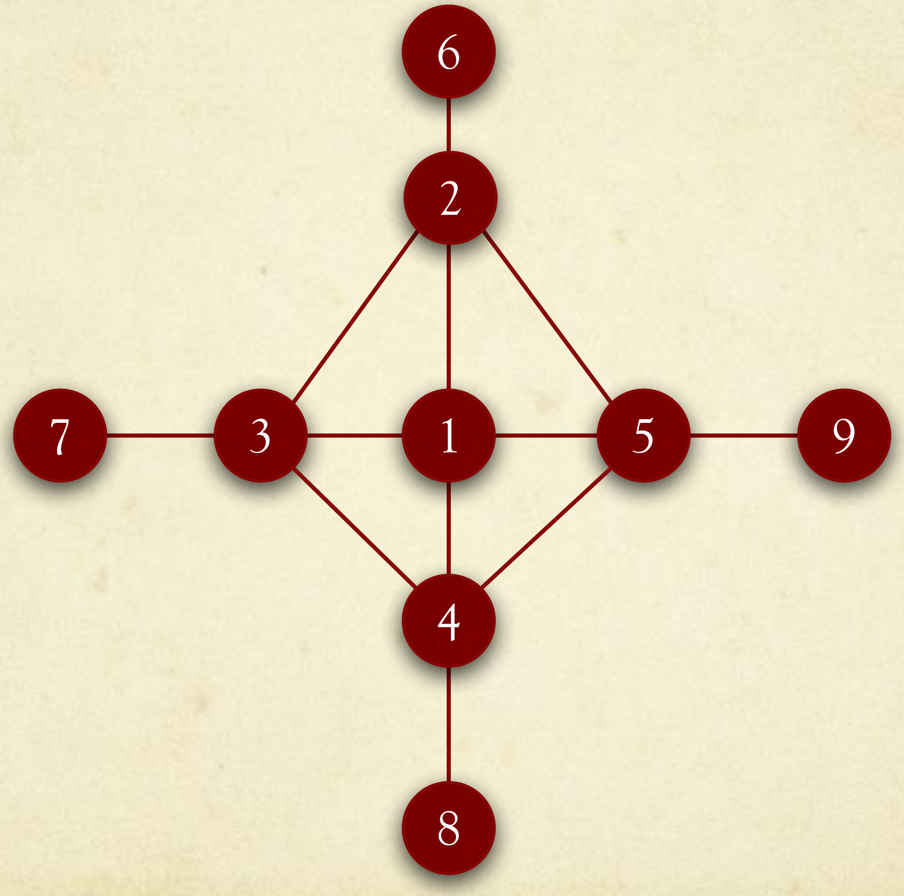
Busca em profundidade (Depth-First Search - DFS)

- Busca em profundidade(Depth-First Search - DFS) explora todos os níveis de cada adjacência, uma por vez
- A partir do vértice inicial, explora-se todos os níveis possíveis de uma adjacência;
- Quando não for mais possível expandir a busca, retorna-se ao último vértice com adjacências ainda não exploradas e retoma-se o processo

DFS

- Uma estrutura de pilha é utilizada para guiar os passos da busca;
- Durante a exploração, um vértice é descoberto na primeira vez em que é encontrado, quando então é empilhado
- Quando o vértice não possui mais adjacências a serem exploradas, ele é desempilhado, sendo considerado terminado.

DFS



http://www.youtube.com/watch?v=dclwa4yMYRo

Desafio

- Implemente um grafo representando os caminhos da sua sala de aula até o bar. Implemente, utilizando uma matriz de adjacência, um algoritmo que encontre pelo menos um caminho possível no grafo.

Revisão

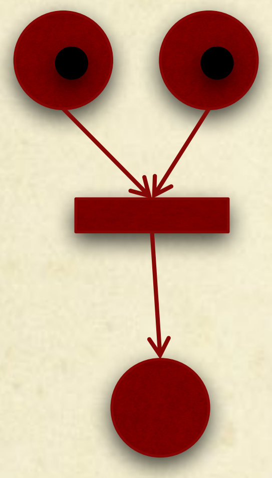
- Grafos são modelos matemáticos que representam ligações entre coisas.
- Podem ser representados em modelos computacionais
- Existem vários algoritmos de busca, onde estudamos dois: Profundidade e Largura

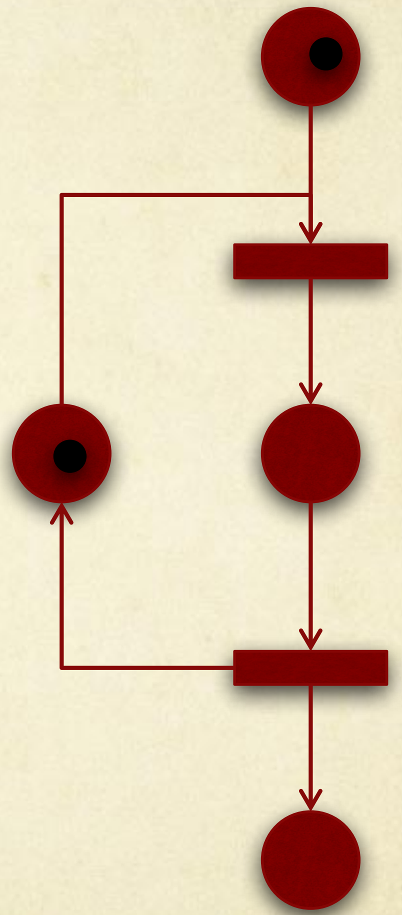
Grafos, um Universo mais abrangente

Tema referido é extremamente extenso, ainda existem diversos algoritmos e tipos de grafos a serem explorados.

- Alguns tipos de grafos foram desenvolvidos para aplicações bem específicas como Grafos em Y, para modelar transformações de contexto, ou Redes de Petry, as quais são bastante utilizadas em Sistemas de informação de processos complexos.

Redes de Petri





Trabalho

- Cada grupo deverá pesquisar pelo menos três aplicações ou soluções implementadas com grafos, mostrando à turma.

Bibliografia

- CORMEN, Thomaz H., LEISERSON, Charles E., RIVEST, Ronald, L., STEIN, Clifford. Introduction to algorithms. Prentice Hall, 2001.
- ZIVIANI, Nivio. Projeto de Algoritmos: Com Implementações em Pascal e C. São Paulo: Pioneira Thomson Learning, 2002.
- AHO, Alfred V., HOPCROFT, John F., UILMAN, Jeffrey D. Data Structure and Algorithms. Massachussetts:
- GOODRICH, Michael T., TAMASSIA, Robert. Estruturas de Dados e Algoritmos em Java.Porto Alegre: Bookman, 2002.

that's all folks!