Algoritmos e Estruturas de Dados III

3º Período Engenharia da Computação

Prof. Edwaldo Soares Rodrigues
Email: edwaldo.rodrigues@uemg.br

UNIVERSIDADE
DO ESTADO DE MINAS GERAIS
UEMG
UNIDADE DIVINÓPOLIS

Grafos

- Introdução:
  - Edsger Dijkstra:
    - “A Ciência da computação tem tanto a ver com o computador como a Astronomia com o telescópio, a Biologia com o microscópio, ou a Química com os tubos de ensaio. A Ciência não estuda ferramentas, mas o que fazemos e o que descobrimos com elas.”;

Grafos

- Introdução:

  - Um grafo é uma estrutura de abstração muito útil na representação e solução de problemas computacionais, por representarem relações de interdependência entre elementos de um conjunto;

Grafos

- Introdução:

  - Em 1735, Euler ganha fama mundial ao resolver um problema que por décadas foi desafio para os matemáticos da época (Série infinita da soma dos inversos dos quadrados;

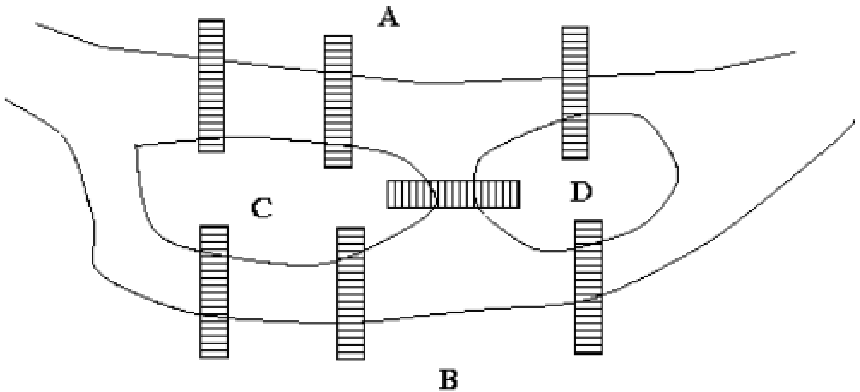
    - Conhecido como problema da Basiléia;

  - A maioria dos grandes matemáticos de seu tempo tentaram sem êxito encontrar o resultado desta série infinita;

  - Euler possuía apenas 28 anos na época;

Grafos

- Introdução:
  - Um ano mais tarde (1736), Euler resolve o problema conhecido como as Sete Pontes de Konigsberg :
    - No século XVIII, havia na cidade de Konigsberg um conjunto de sete pontes que cruzavam o rio Pregel. Elas conectavam duas ilhas (C e D) entre si e as ilhas com as margens (A e B).

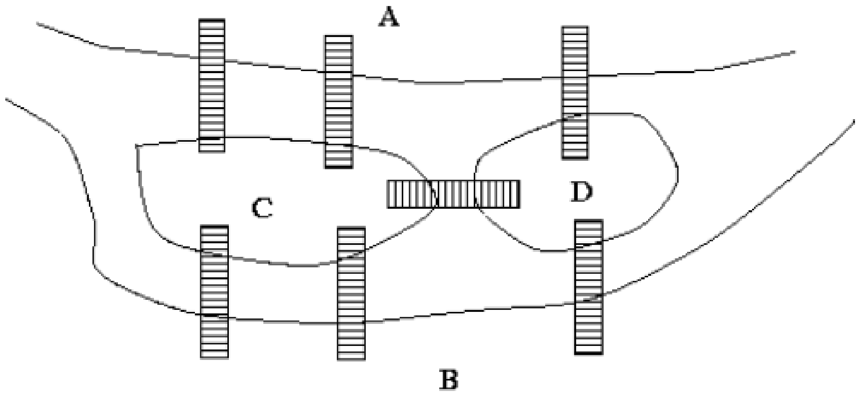


Grafos

- Introdução:

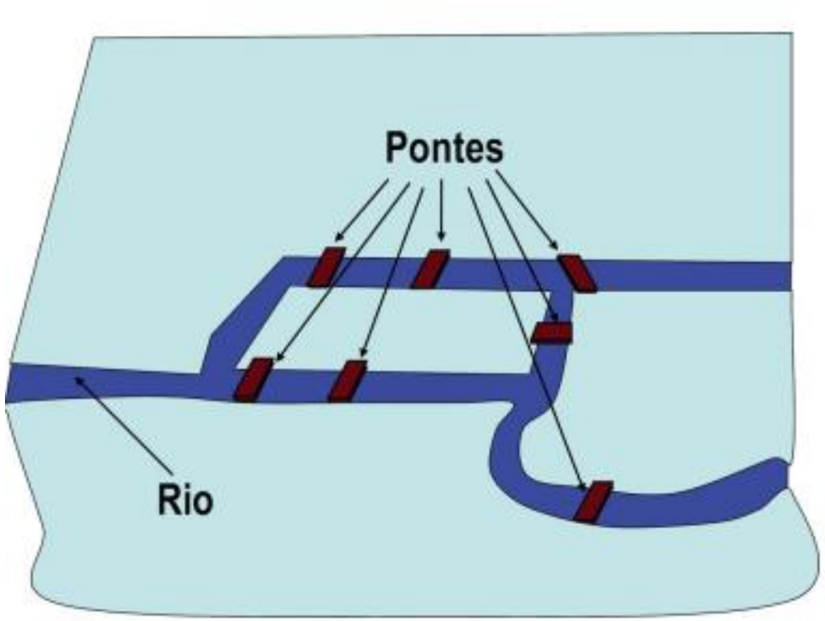
  - Pontes de Konigsberg:

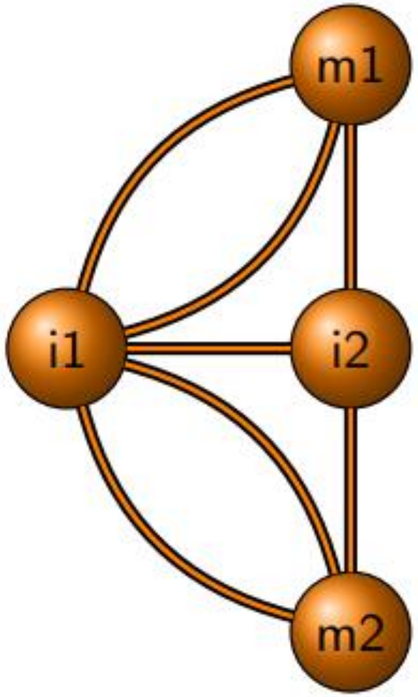
    - Os moradores locais questionavam se seria possível começando em um dos blocos de terra (A, B, C ou D), caminhar exatamente uma única vez sobre cada uma das pontes e retornar ao ponto de partida.



Grafos

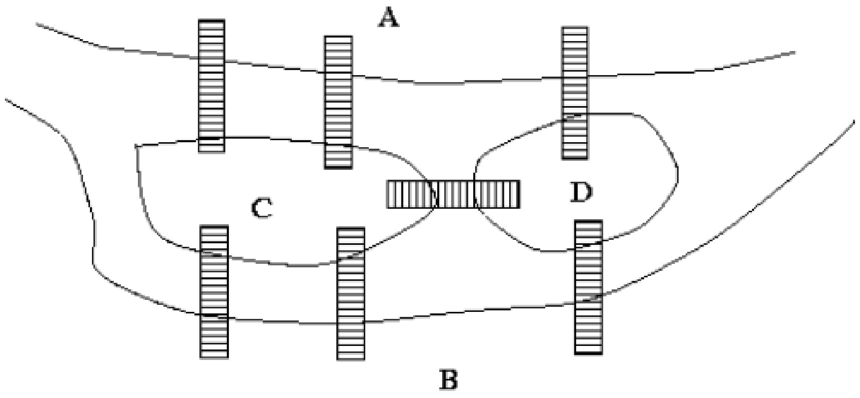
- Introdução:
  - Pontes de Konigsberg:





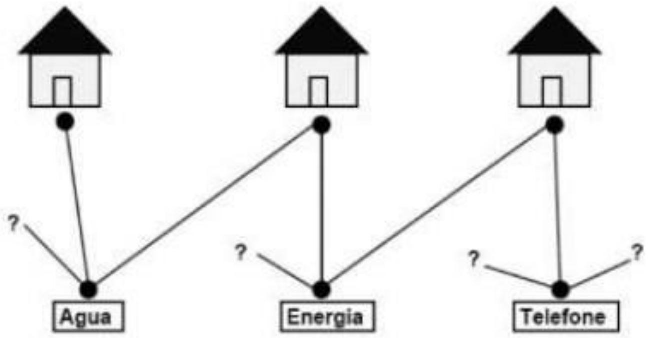
Grafos

- Introdução:
  - Pontes de Konigsberg:
    - Euler provou através de grafos que não é possível realizar tal façanha!



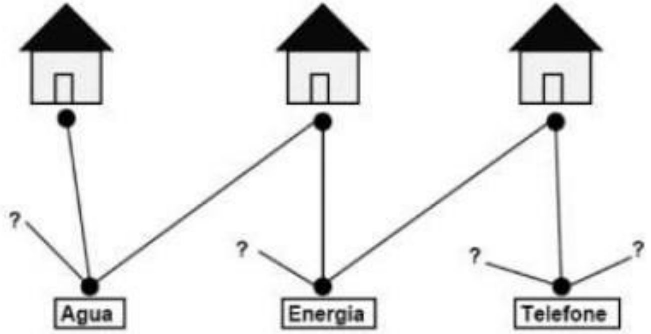
Grafos

- Mais alguns problemas clássicos:
  - O problema das três casas:
    - Suponha que existam três casas e que cada uma delas deve ser conectada a três serviços básicos: água, eletricidade e telefone



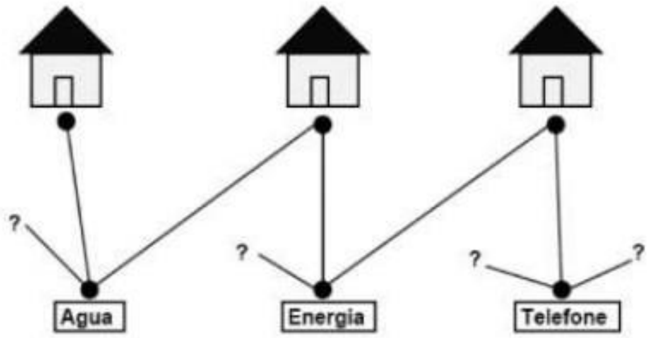
Grafos

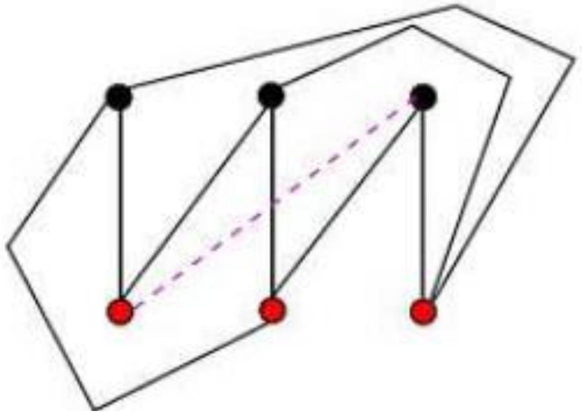
- Mais alguns problemas clássicos:
  - O problema das três casas:
    - Considerando que todos os fios e canos estão no mesmo plano, é possível fazer todas as conexões sem haver cruzamento de tubulação?



Grafos

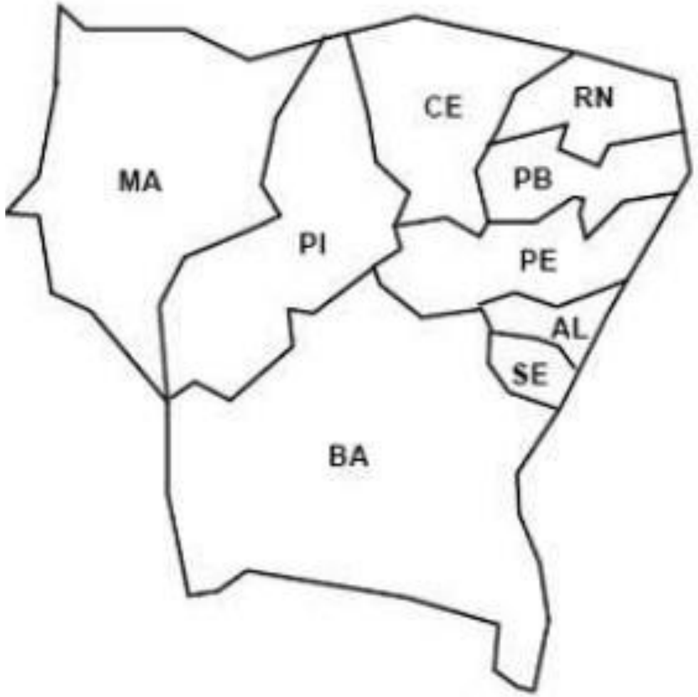
- Mais alguns problemas clássicos:
  - O problema das três casas:
    - Não é possível fazer as conexões sem cruzamentos!





Grafos

- Mais alguns problemas clássicos:
  - O problema da Coloração de Mapas:
    - Considere o mapa da região Nordeste:

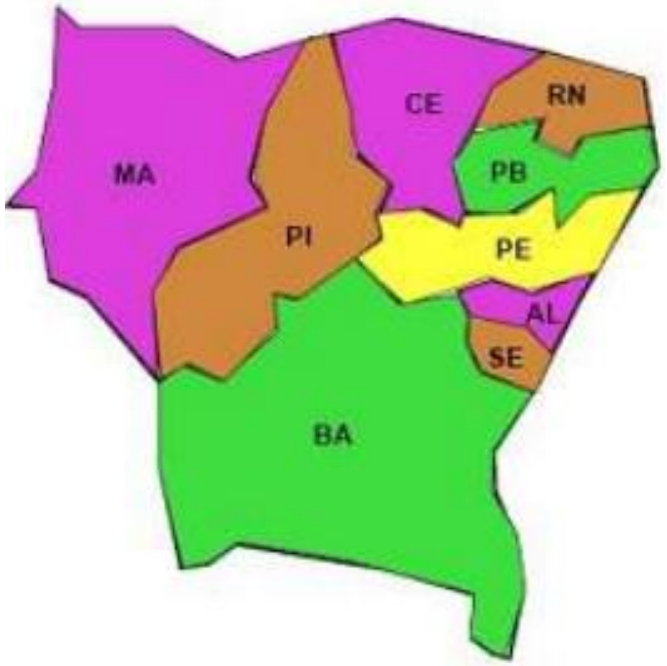


    - Com quantas cores é possível pintar os estados fazendo com que nenhum estado tenha a mesma cor de seus vizinhos?

Grafos

- Mais alguns problemas clássicos:

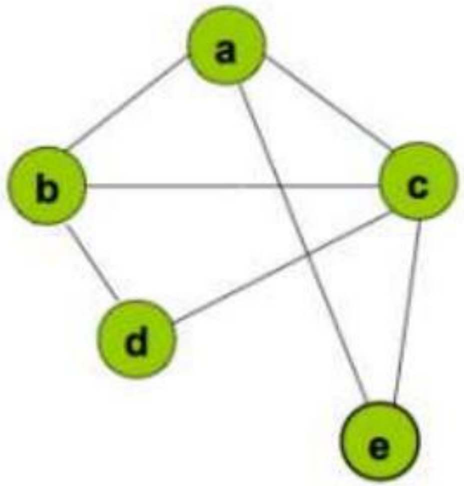
  - O problema da Coloração de Mapas:



  - Com quantas cores é possível pintar os estados fazendo com que nenhum estado tenha a mesma cor de seus vizinhos?
    - R: 4 cores;

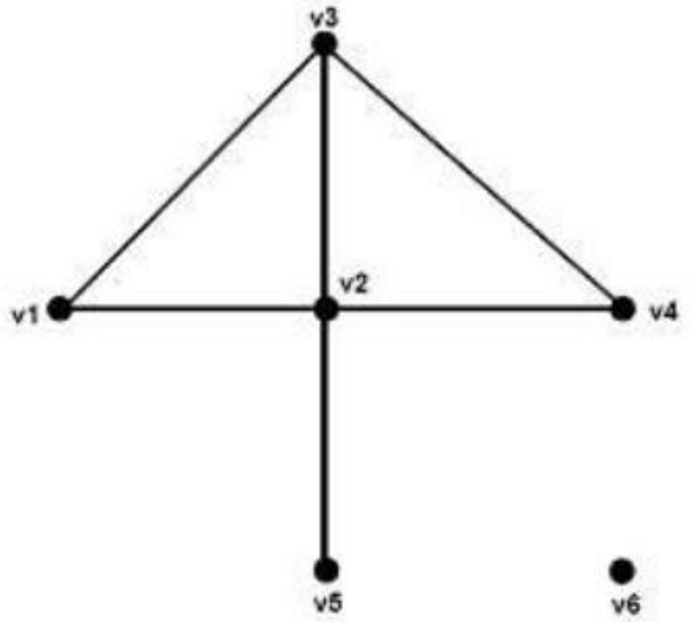
Grafos

- Definição formal:
  - Grafo G = (V, A)
  - Conjunto V com n vértices (também chamados nós)
    - $\{v_1, v_2, \ldots, v_n\}$
  - Conjunto A com m arestas ou arcos
    - $\{a_1, a_2, \ldots, a_m\}$



Grafos

- Definição formal:
  - Exemplo de grafo: G = (V, A):
    - V = {v1, v2, v3, v4, v5, v6};
    - A = {(v1, v2), (v1, v3), (v2, v3), (v2, v4), (v2, v5), (v3, v4)};

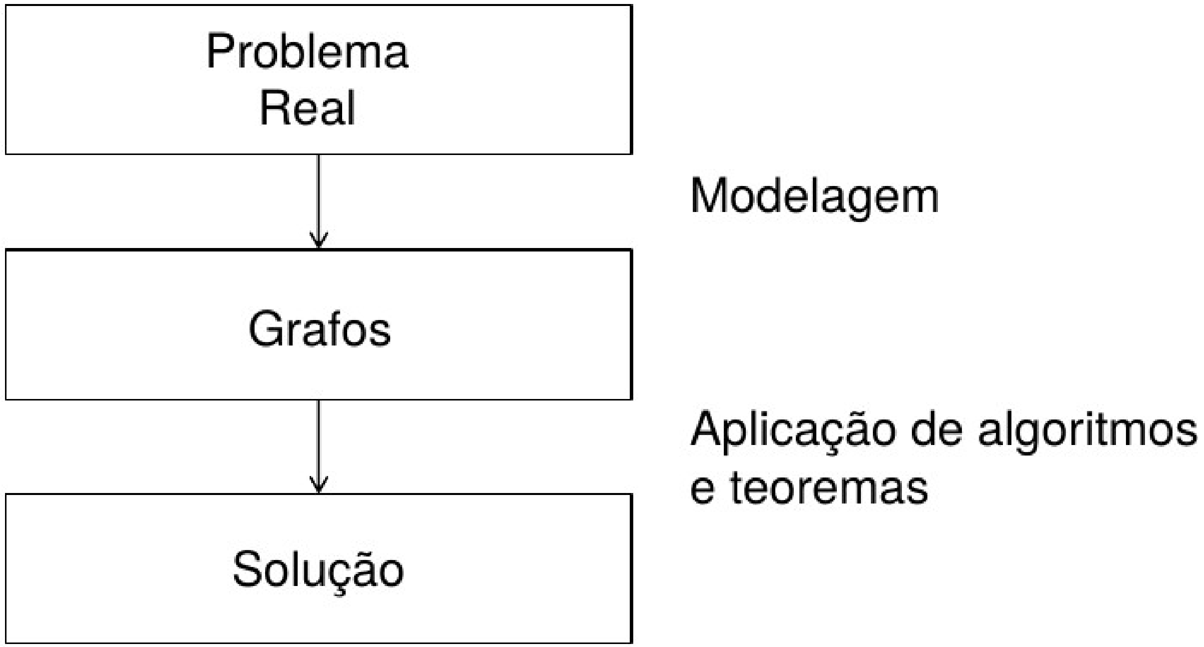


Grafos

- Por que estudar grafos?
  - Importante ferramenta matemática com diversas aplicações:
    - Um grande número de problemas pode ser representado como problemas em grafos;
    - Possui diversas soluções prontas para uso;
  - A maior dificuldade está em expressar (modelar) o problema em grafos;

Grafos

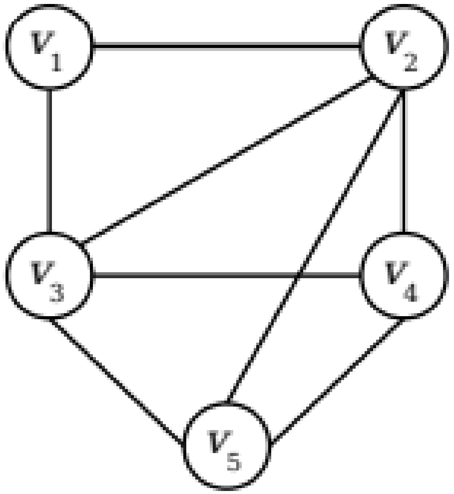
- Processo de abstração:



Grafos

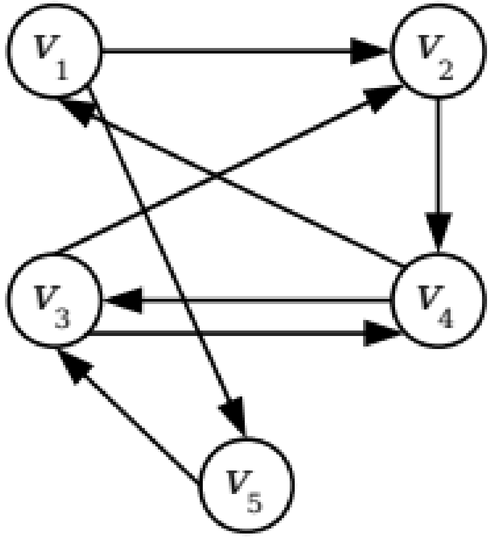
- Grafo Não Direcionado (GND):

  - Ligações expressas em Arestas
  - Se o vértice *a* está ligado a *b*, a recíproca é verdadeira;
  - Cada aresta é representada por um conjunto {v1, v2} , indicando os dois vértices envolvidos.



Grafos

- Grafo Direcionado (GD):
  - Ligações expressas em Arcos →;
  - Cada arco é representada por um par ordenado (v1, v2), indicando os dois vértices envolvidos;

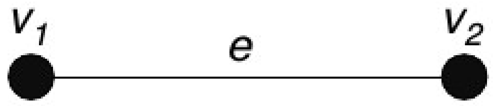


Grafos

- Incidência:

  - Uma aresta e = (V1, V2) é dita incidente a v1 e a v2;

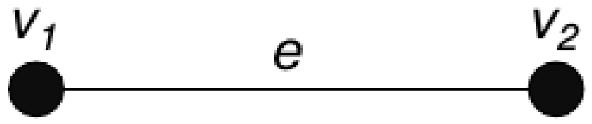
  - Os vértices que a aresta conecta também são ditos incidentes à aresta;



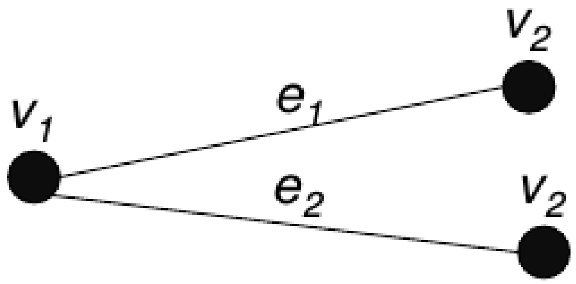
Grafos

- Adjacência:

    - Dois vértices que são incidentes a uma mesma aresta são ditos adjacentes:

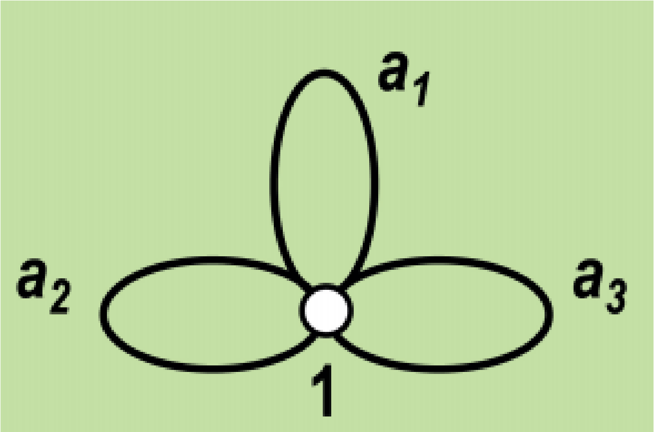


    - Duas arestas que são incidentes a um mesmo vértice são ditas adjacentes:



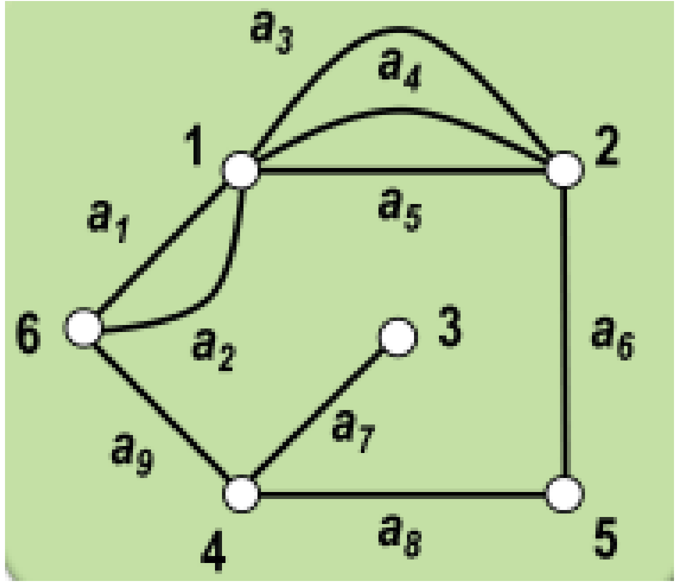
Grafos

- Terminologia:
  - Laço: uma aresta cujas duas extremidades incidem em um mesmo vértice;



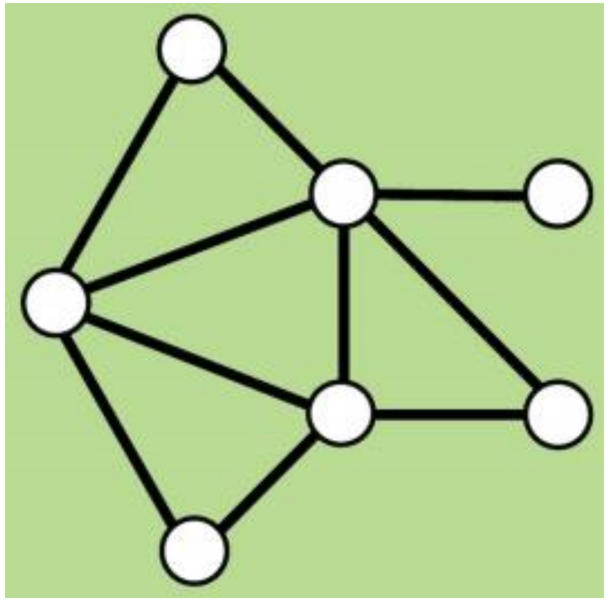
Grafos

- Terminologia:
  - Arestas paralelas: mais de uma aresta associada ao mesmo par de vértices



Grafos

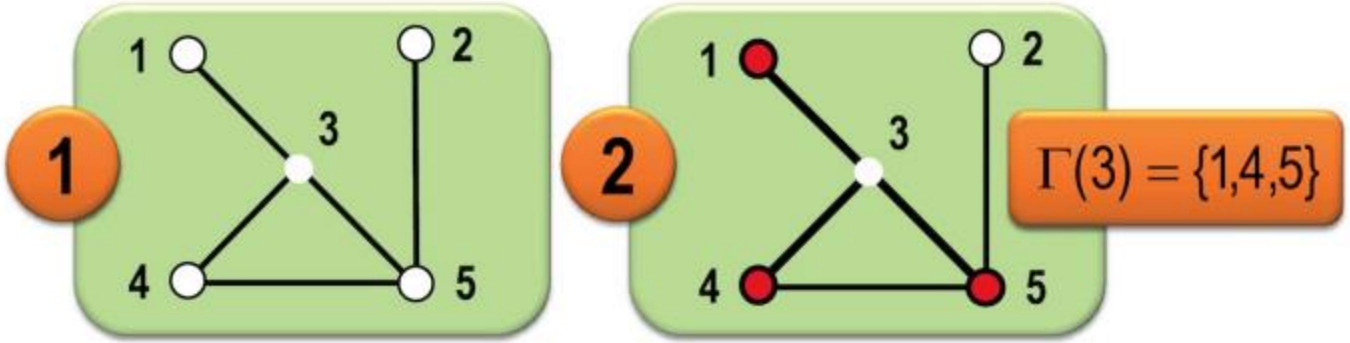
- Terminologia:
  - Grafo Simples: grafo que não possui laços e nem arestas paralelas;



Grafos

- Terminologia:

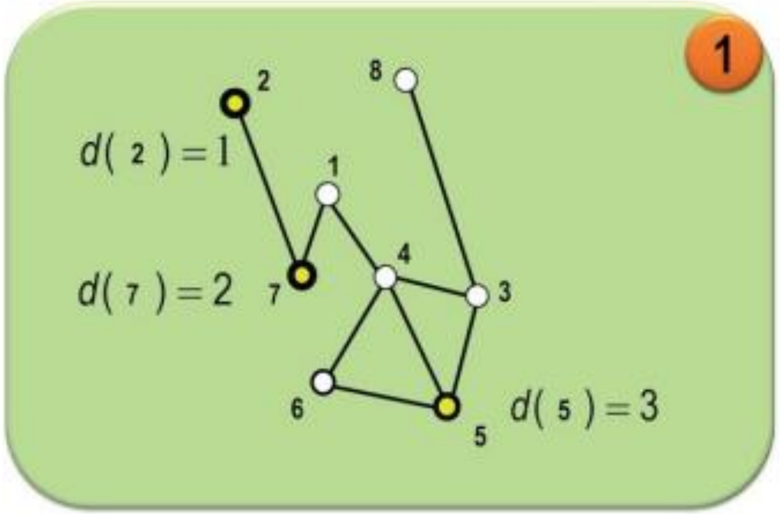
  - Vértices Adjacentes: vértices que são os pontos finais de uma mesma aresta;

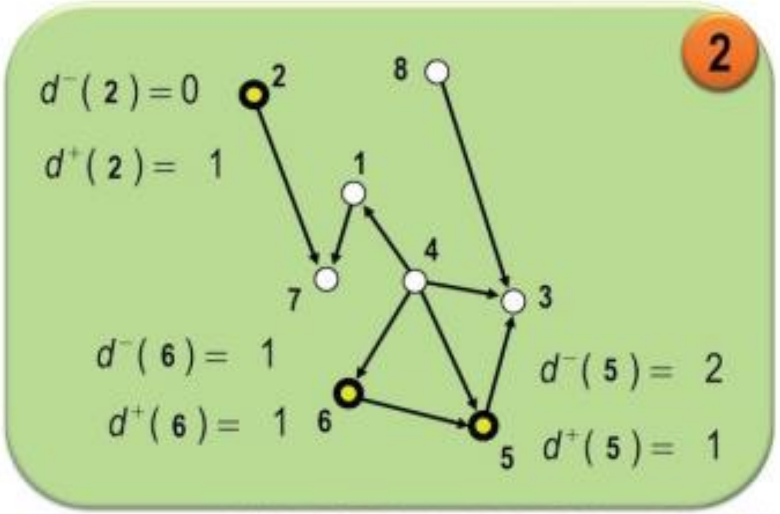


Grafos

- Terminologia:

  - Grau de um vértice:
    - O grau (d(i)) de um vértice i em um grafo não direcionado é igual o número de arestas incidentes a i;
    - O grau de entrada (d−(i)) de um vértice i em um grafo direcionado é igual o número de arestas que entram em i;
    - O grau de saída (d+(i)) de um vértice i em um grafo direcionado é igual o número de arestas que saem de i;





Grafos

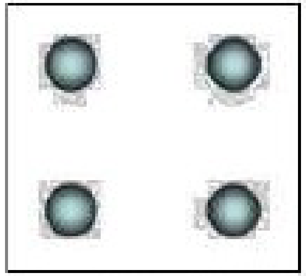
- Conceito:
  - Teorema do Aperto de Mãos Handshaking:
    - A soma dos graus de todos os vértices de um GND G é duas vezes o número de arestas de G;

$$\sum_{i=1}^{n} d(i) = 2m$$

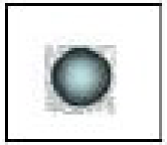
    - O número de vértices de grau ímpar em um GND é par;

Grafos

- Terminologia:
  - Vértice isolado:
    - Vértice de grau 0;
  - Vértice pendente:
    - Vértice de grau 1;
  - Grafo nulo:
    - Grafo sem nenhuma aresta;
  - Grafo trivial:
    - Grafo de apenas 1 vértice e nenhuma aresta;



$|E| = \emptyset$
(grafo nulo)

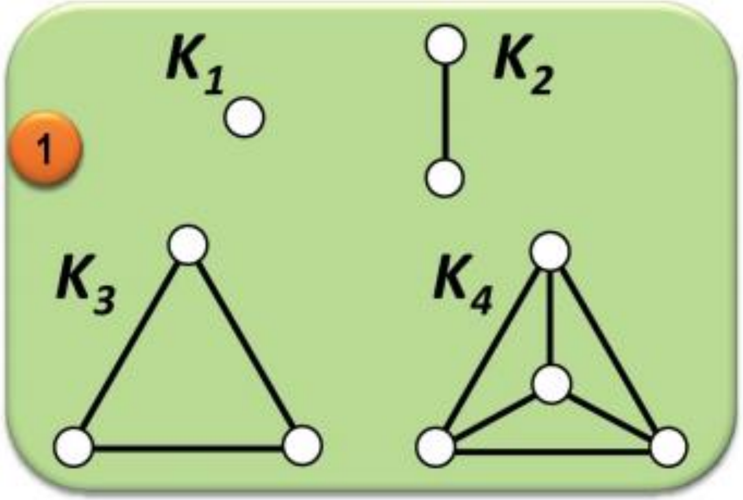


$|V| = 1$ e $|E| = \emptyset$
(grafo trivial)

Grafos

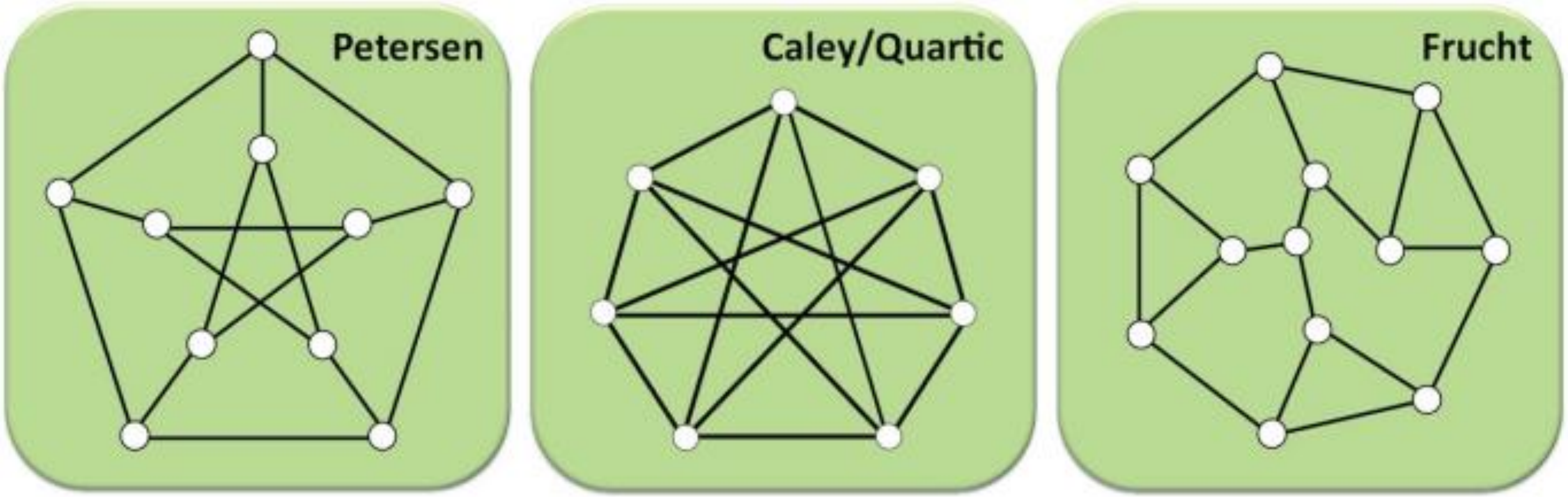
- Terminologia:

  - Grafo Completo: um grafo completo com n vértices, denominado K_n é um grafo simples contendo exatamente uma aresta para cada par de vértices distintos;



Grafos

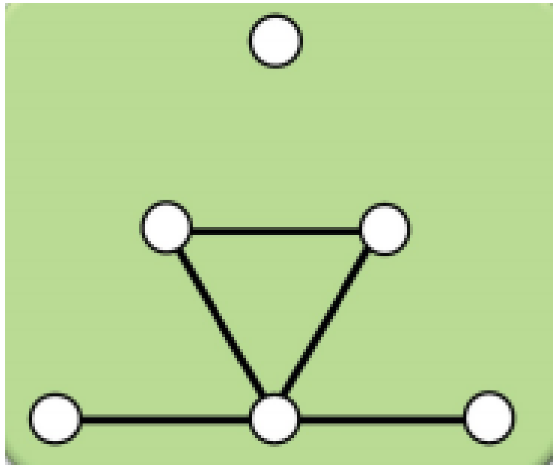
- Terminologia:
  - Grafo Regular: grafo no qual todos os vértices possuem o mesmo grau;
  - Obs: qualquer grafo completo é regular;



Grafos

- Terminologia:

  - Vértice Isolado: vértice com nenhuma aresta incidente;

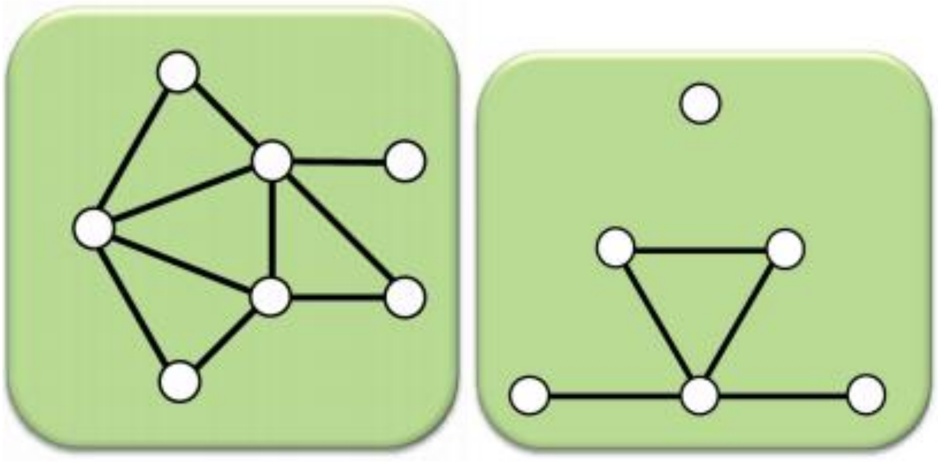


Grafos

- Terminologia:

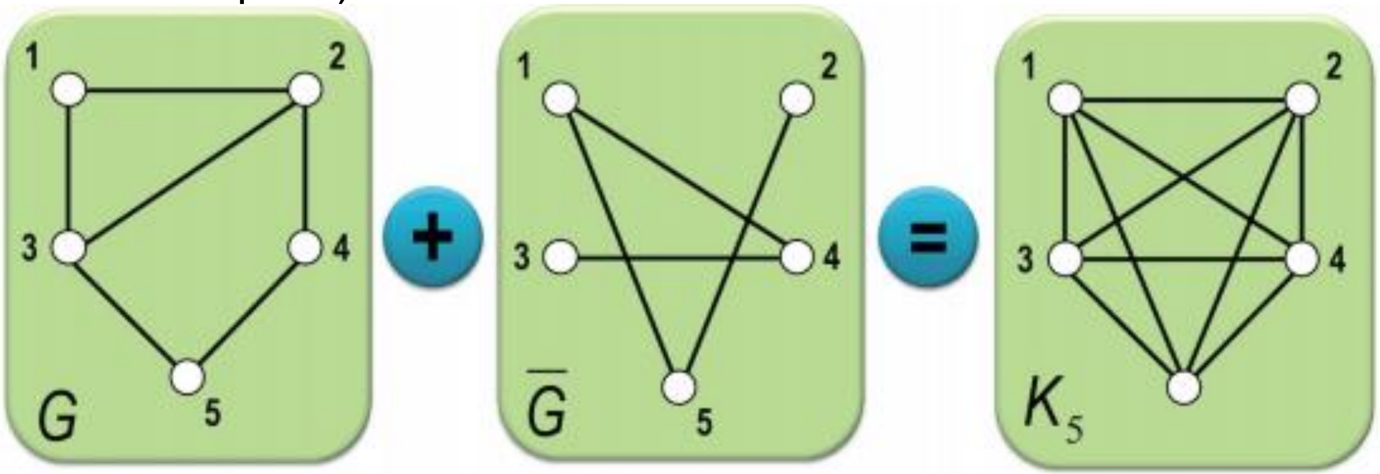
  - Grafo Conexo: Para todo par de vértices i e j de G existe pelo menos um caminho entre i e j;

  - Grafo Desconexo: Consiste de 2 ou mais grafos conexos, chamados de componentes;



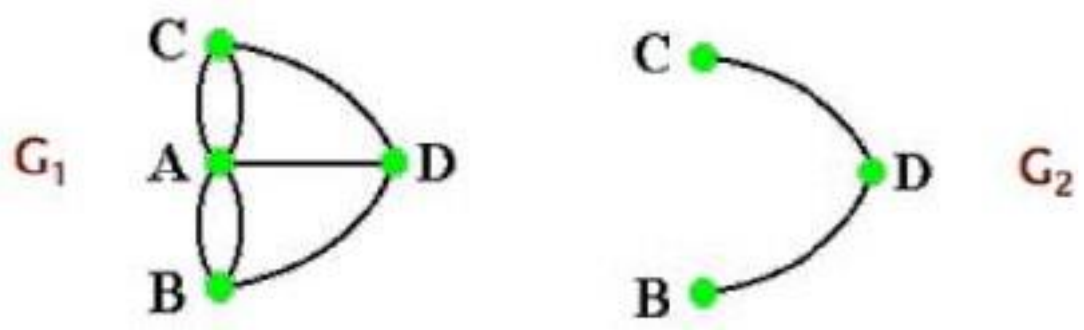
Grafos

- Terminologia:
  - Grafo Complemento:
    - Seja G = (V, A) um grafo simples dirigido ou não-dirigido, o complemento de G, $\bar{G}$ (ou C(G)), é um grafo formado da seguinte maneira:
      - Os vértices de $\bar{G}$ são todos os vértices de G;
      - As arestas de $\bar{G}$ são exatamente as arestas que faltam em G para formarmos um grafo completo;



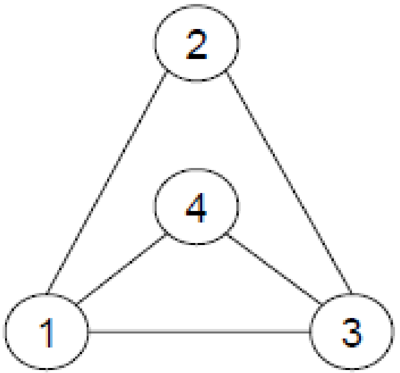
Grafos

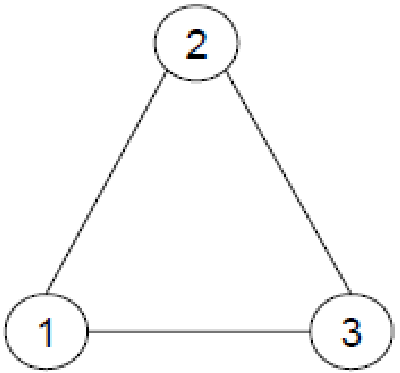
- Tipos de grafos:
  - Subgrafos:
    - Dois grafos G1 e G2 são iguais se V1 = V2 e A1 = A2;
    - Um grafo GB é um subgrafo de G se e somente se os conjuntos de vértices e arestas de GB estão contidos nos conjuntos de vértices e arestas de G;



Grafos

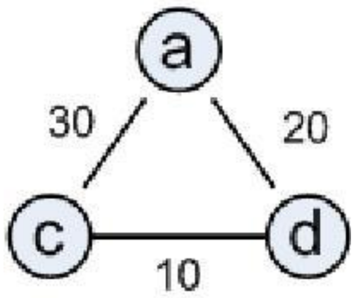
- Tipos de grafo:
  - Clique:
    - O clique de um grafo é um subgrafo de G que seja completo;





Grafos

- Tipos de grafo:
  - Grafos valorados:
    - São utilizados rótulos também nas arestas;



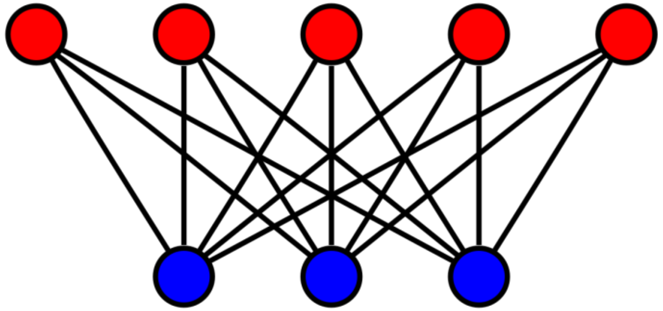
    - Grafos valorados podem ser direcionados e não direcionados;

Grafos

- Tipos de grafo:
  - Grafos valorados:
    - Geralmente utilizamos rótulos em arestas para representar o custo de alguma coisa;
    - Por exemplo, a distância para sair da cidade a e chegar na cidade b;
    - Ou o tempo necessário;
    - Em redes de computadores, a aresta muitas vezes recebe o RTT (round-trip time), tempo de ida e volta;

Grafos

- Tipos de grafo:
  - Grafos bipartidos:
    - Um grafo G(V,A) é chamado de grafo bipartido se o seu conjunto de vértices puder ser dividido em dois subconjuntos V_1 e V_2 sem intersecção;
    - E as arestas conectam apenas os vértices que estão em subconjuntos diferentes, ou seja, uma aresta sempre conecta um vértice de V_1 a V_2 ou vice-versa, porém ela nunca conecta vértices do mesmo subconjunto entre si;

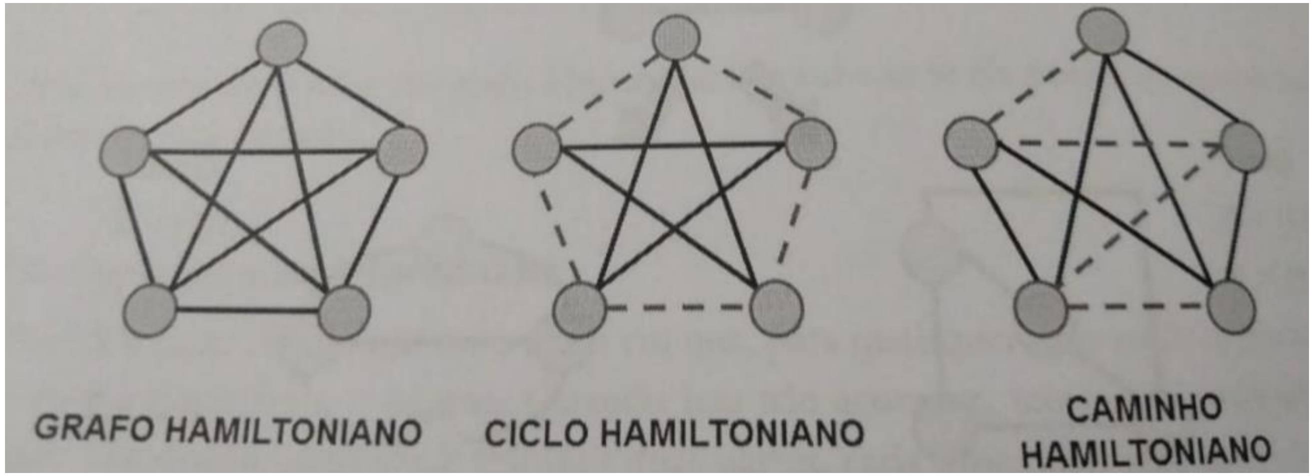


Grafos

- Tipos de grafo:

  - Grafos Hamiltonianos:

    - Um grafo hamiltoniano é um tipo especial de grafo que possui um caminho que visita todos os seus vértices apenas uma vez;
    - A esse caminho, dá-se o nome de caminho hamiltoniano;
    - Um ciclo hamiltoniano é um ciclo no qual cada vértice é visitado exatamente uma vez, retornando ao seu ponto de partida (esse é o único vértice que se repete).

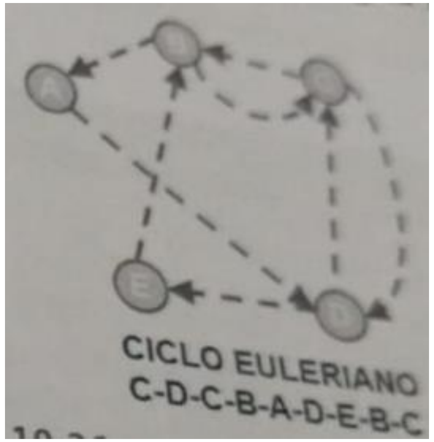


Grafos

- Tipos de grafo:

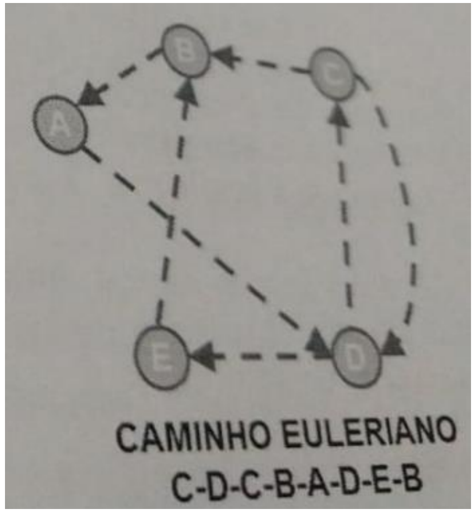
  - Grafos Eulerianos:

    - Um grafo euleriano é um tipo especial de grafo que possui um ciclo que visita todas as suas arestas apenas uma vez, iniciando e terminando no mesmo vértice;
    - A esse ciclo dá-se o nome de ciclo euleriano;



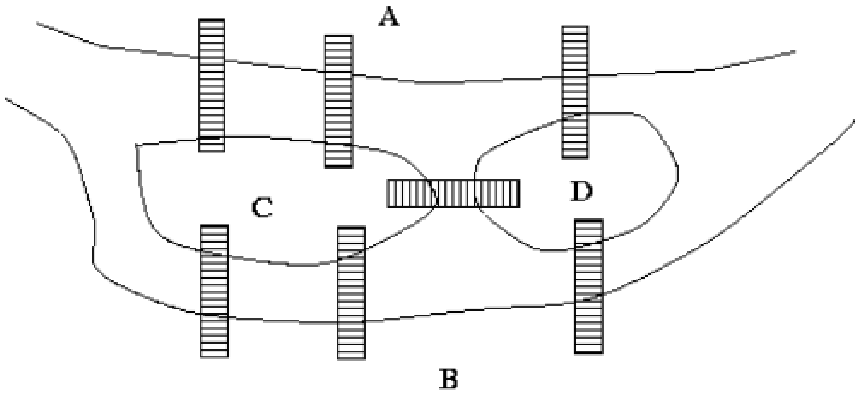
Grafos

- Tipos de grafo:
  - Grafos Semieulerianos:
    - Um grafo semieuleriano é um tipo especial de grafo que possui um caminho aberto (não é um ciclo) que visita todas as suas arestas apenas uma vez;
    - A esse caminho dá-se o nome de caminho euleriano;



Grafos

- Pontes de Konigsberg – Voltando ao problema:
  - Os moradores locais questionavam se seria possível começando em um dos blocos de terra (A, B, C ou D), caminhar exatamente uma única vez sobre cada uma das pontes e retornar ao ponto de partida.
  - Vimos anteriormente que Euler chegou a conclusão de que não era possível, mas porque?



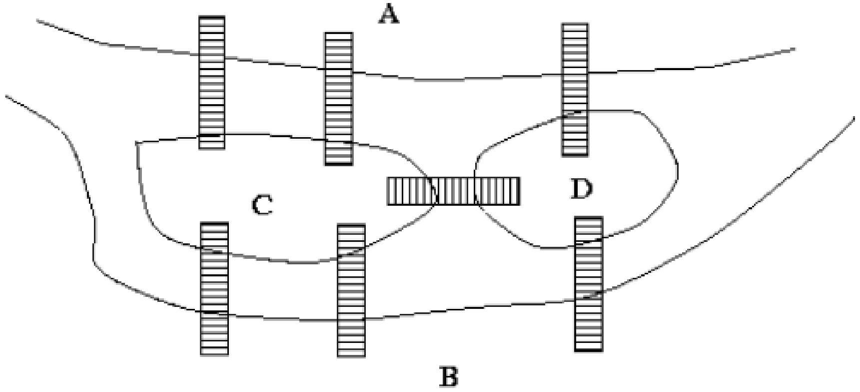
UNIVERSIDADE
DO ESTADO DE MINAS GERAIS
UEMG
UNIDADE DIVINÓPOLIS

Grafos

- Pontes de Konigsberg – Voltando ao problema:

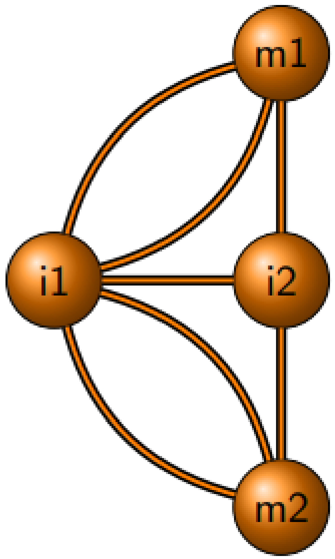
  - Partindo do vértice A, e percorrendo outros vértices, podemos ver a utilização de no mínimo duas arestas (pontes) “chegada” e a de “saída”;

  - Assim, se for possível achar uma rota que usa todas as arestas do grafo e começa e termina em A, então o número total de “chegadas” e “saídas” de cada vértice deve ser um valor múltiplo de 2;



Grafos

- Pontes de Konigsberg – Voltando ao problema:



  - No entanto, temos:
    - Grau(A) = Grau(C) = Grau(D) = 3;
    - Grau(B) = 5;

  - Assim, por este raciocínio não é possível percorrer as faixas de terra passando por cada ponte uma única vez, retornando ao vértice de partida;

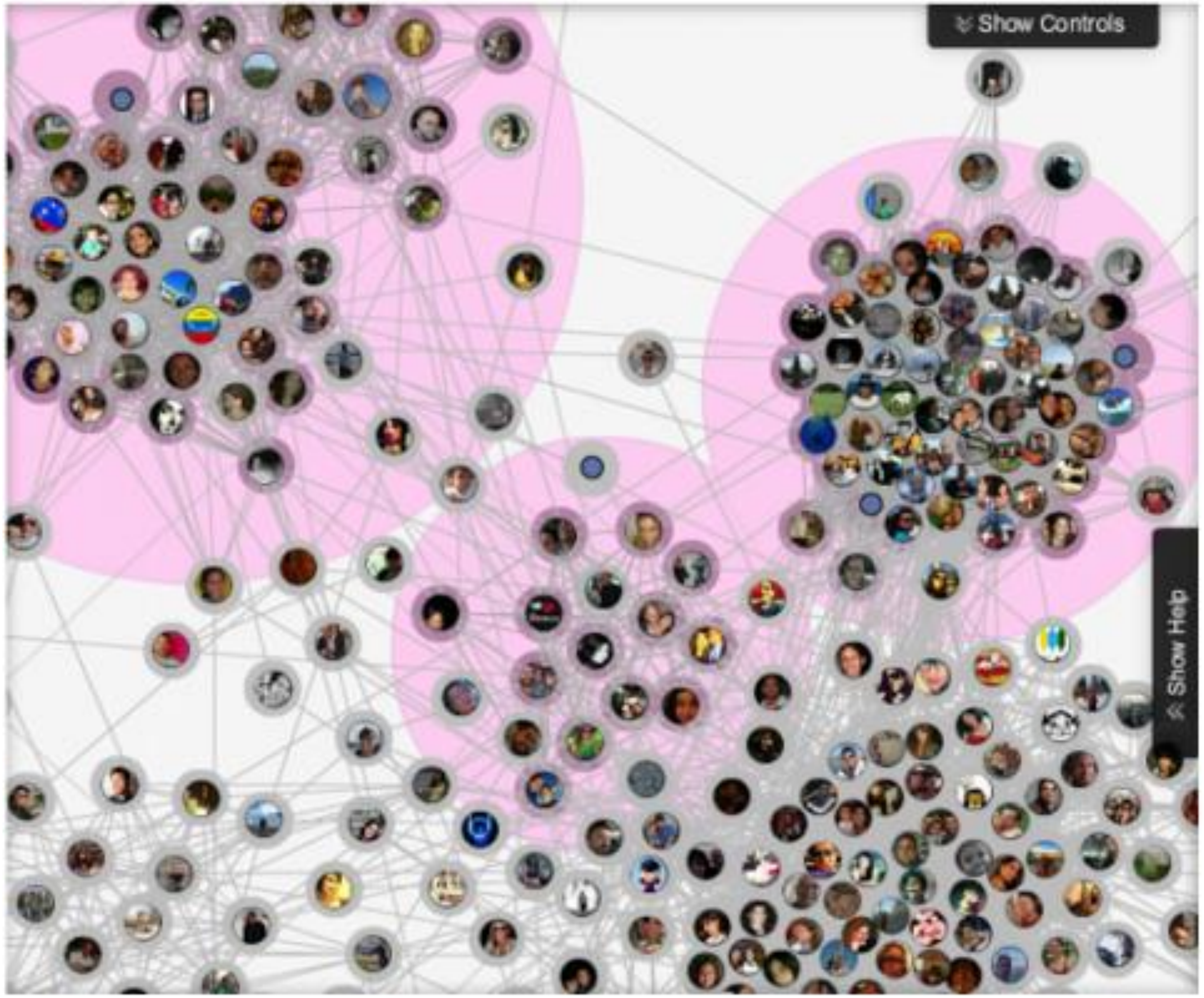
Grafos

- Aplicações:

Facebook: Fevereiro de 2017, ≈ 2,13 bilhão de usuários ativos.

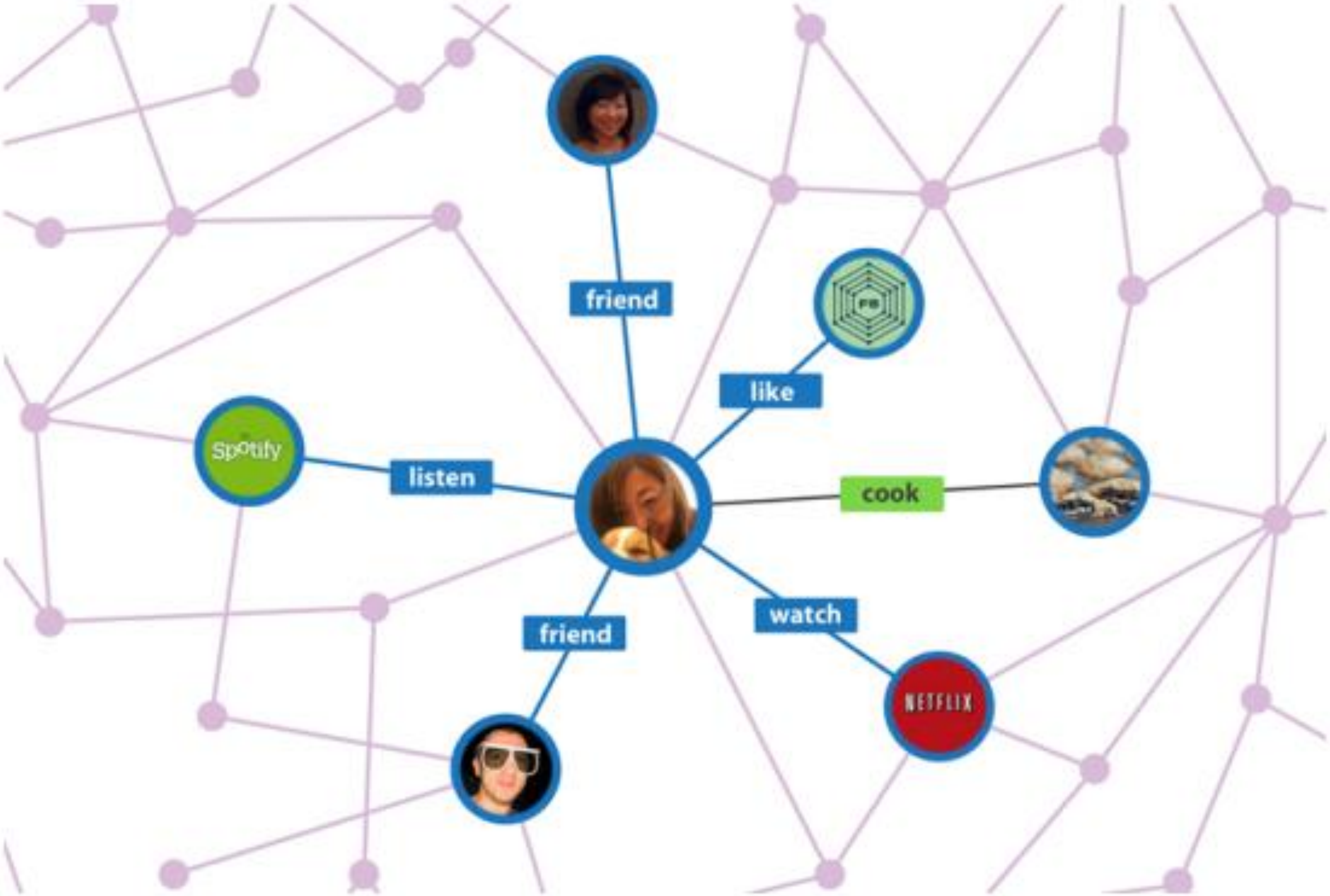
Grafos

- Aplicações:



Grafos

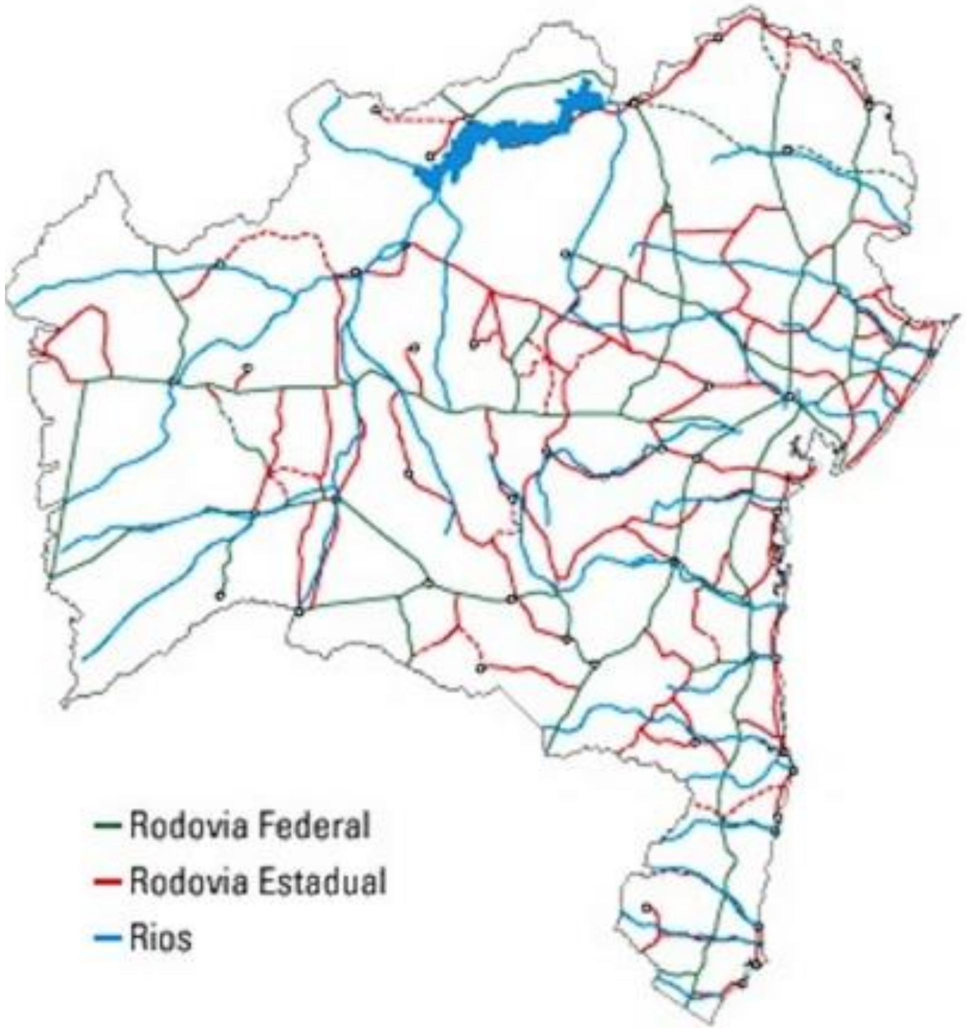
- Aplicações:



Facebook Graph Search.

Grafos

- Aplicações:

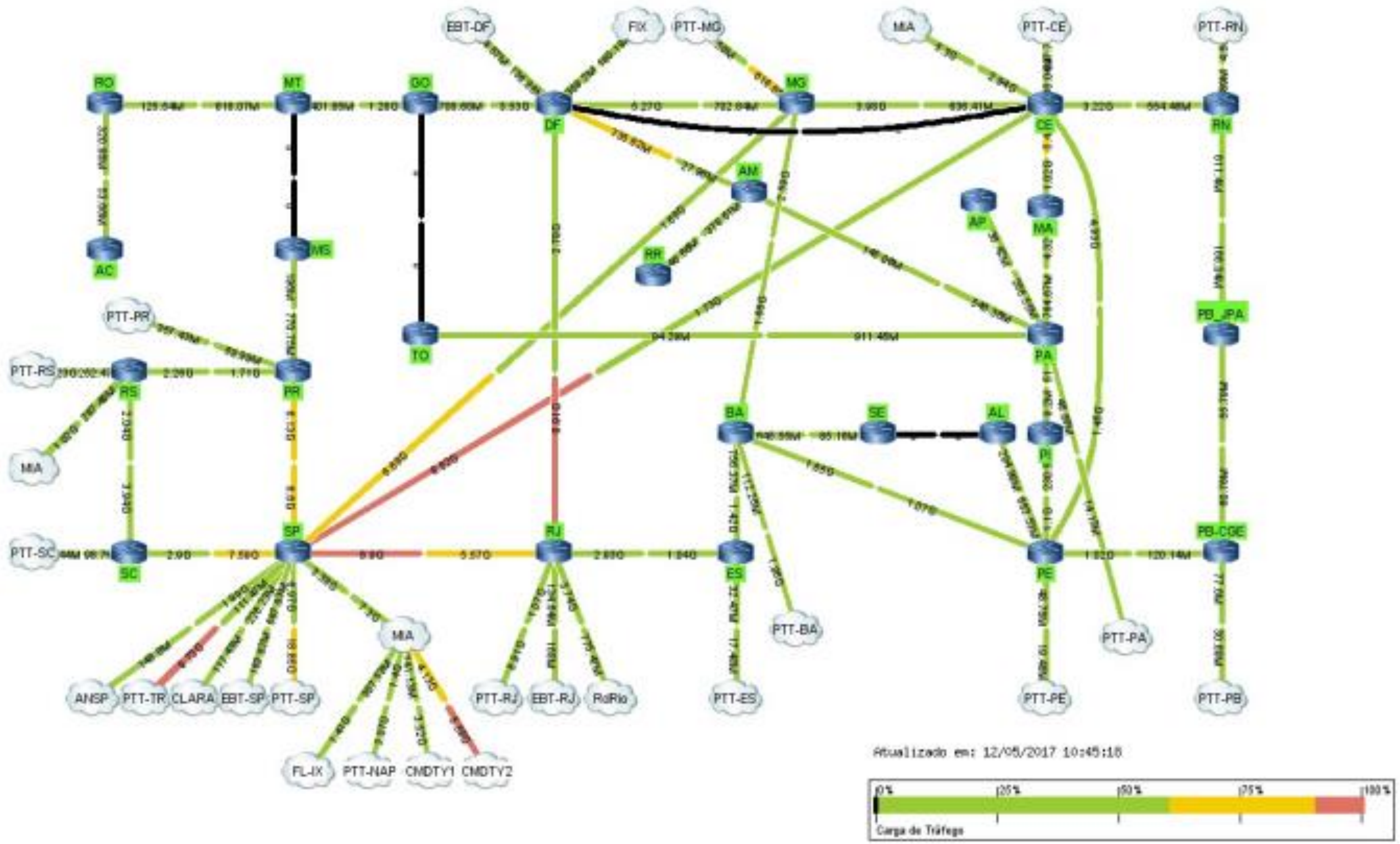


Grafos

- Aplicações:

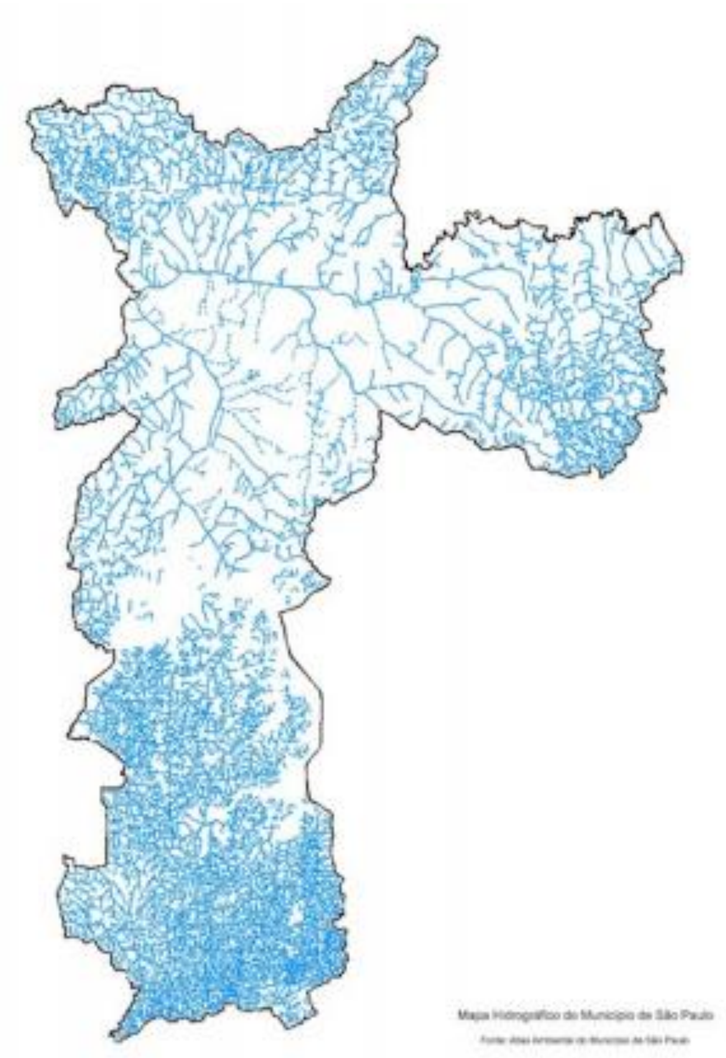
Grafos

- Aplicações:



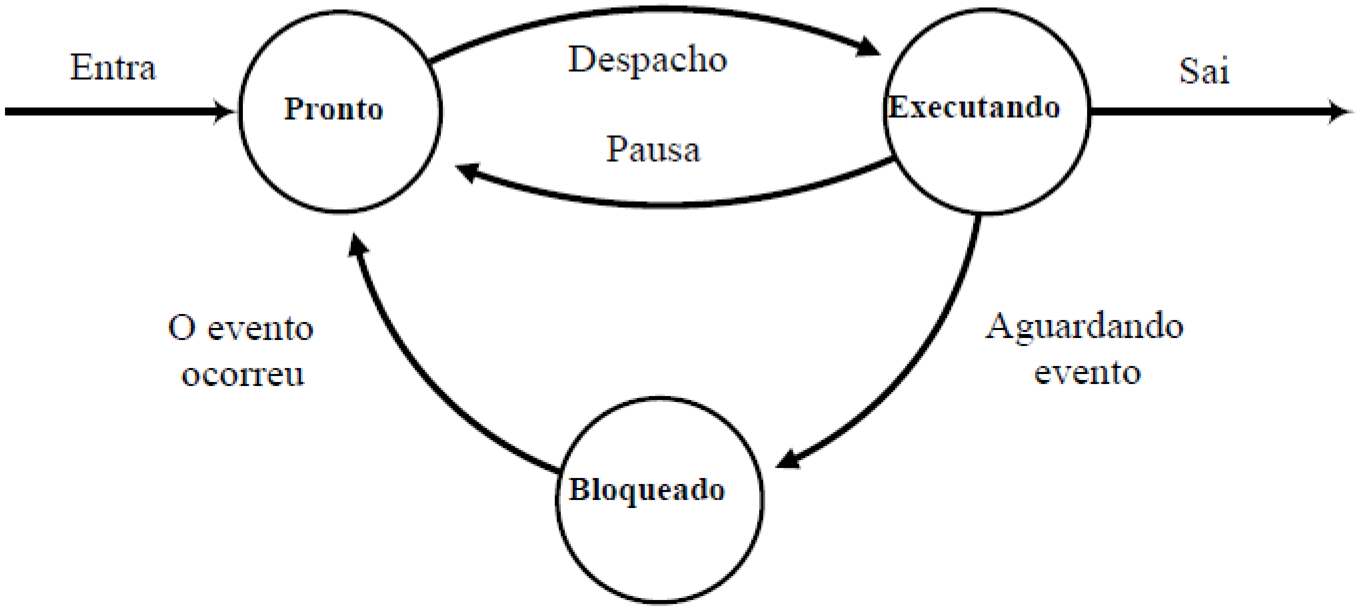
Grafos

- Aplicações:



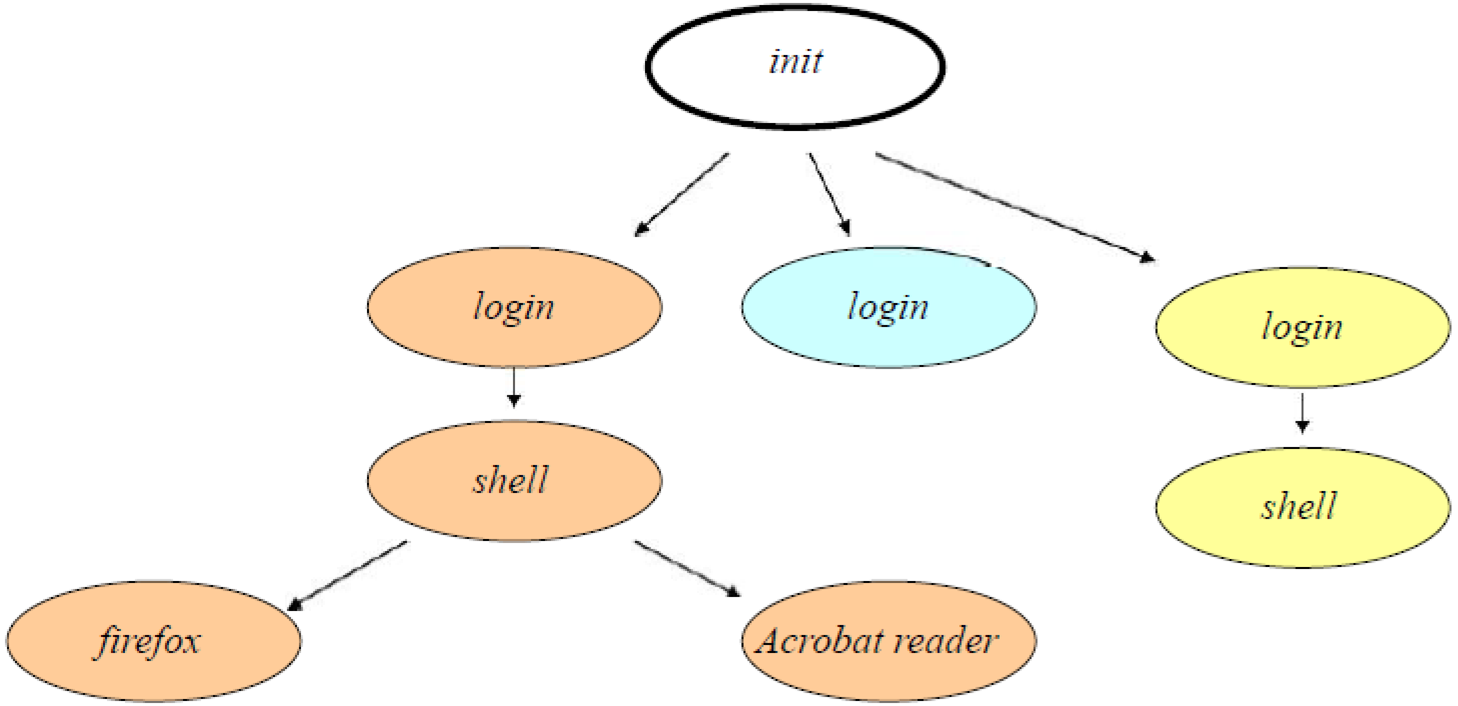
Grafos

- Aplicações:
  - Sistemas Operacionais: abstraindo os estados de processos/threads;



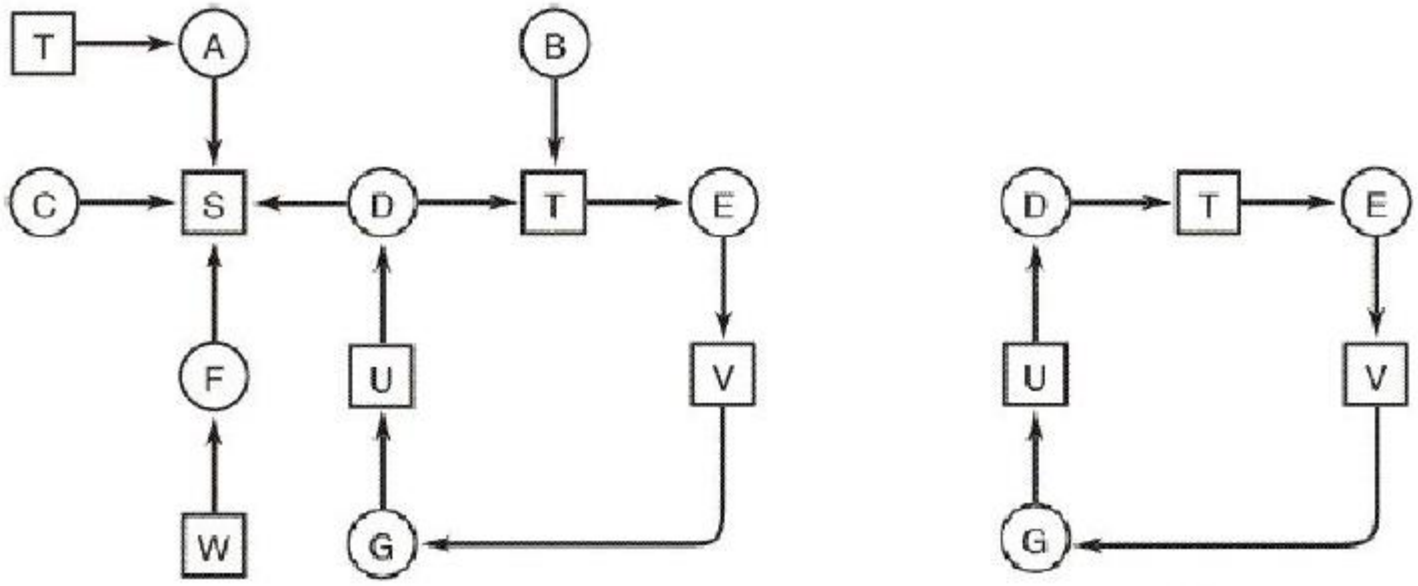
Grafos

- Aplicações:
  - Sistemas Operacionais: Hierarquia de processos – Árvores são grafos especiais;



Grafos

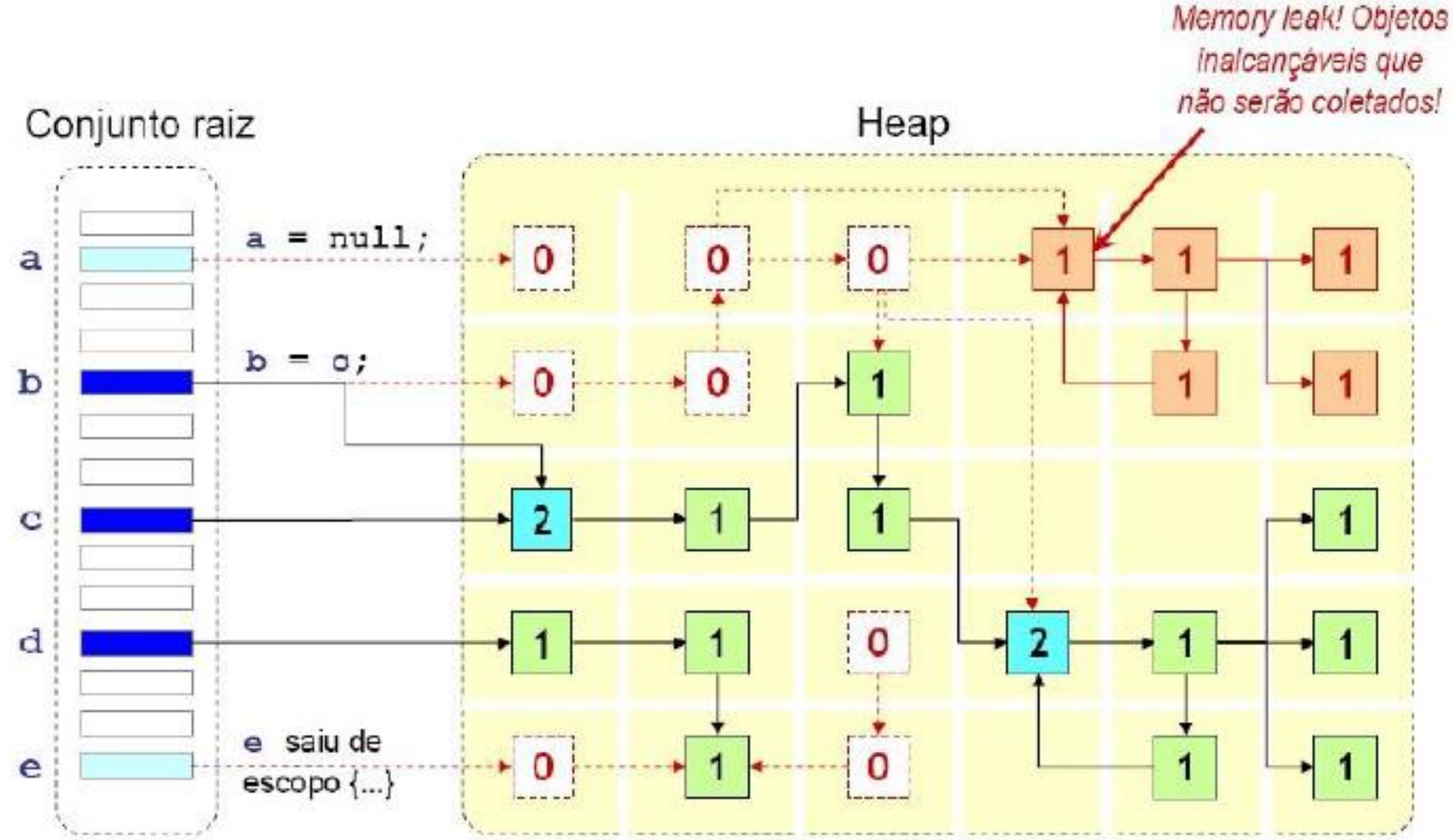
- Aplicações:
  - Sistemas Operacionais: Detecção de deadloacks, por meio de ciclo no grafo;



(a) (b)

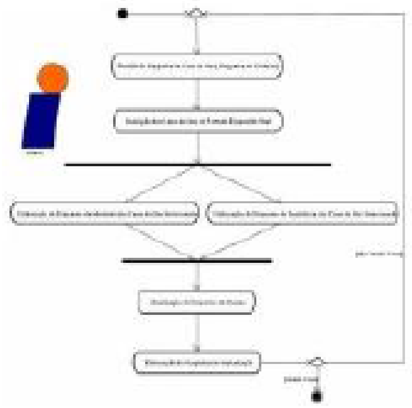
Grafos

- Aplicações:
  - Programação: Garbage collector Java;

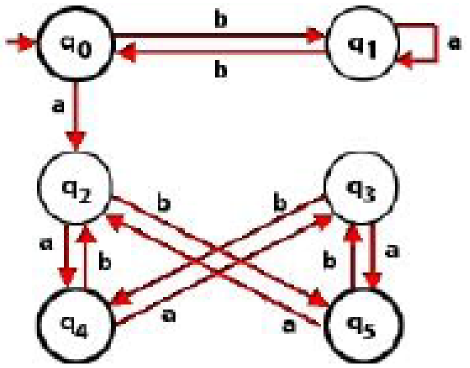


Grafos

- Aplicações:
  - Teoria da computação e Engenharia de Software;



Um requisito gera um diagrama de estados (UML)



Um autômato

Algoritmos e Estruturas de Dados III

- Bibliografia:
  - Básica:

    - ASCENCIO, Ana C. G. Estrutura de dados. Rio de Janeiro: Pearson. 2011.
    - CORMEN, Thomas; RIVEST, Ronald; STEIN, Clifford; LEISERSON, Charles. Algoritmos. Rio de Janeiro: Elsevier, 2002.
    - ZIVIANI, Nívio. Projeto de algoritmos com implementação em Pascal e C. São Paulo: Cengage Learning, 2010.

  - Complementar:

    - EDELWEISS, Nina, GALANTE, Renata. Estruturas de dados. Porto Alegre: Bookman. 2009. (Coleção Livros didáticos de informática UFRGS, 18).
    - PINTO, W.S. Introdução ao desenvolvimento de algoritmos e estrutura de dados. São Paulo: Érica, 1990.
    - PREISS, Bruno. Estruturas de dados e algoritmos. Rio de Janeiro: Campus, 2000.
    - TENEMBAUM. Aaron M. Estruturas de Dados usando C. São Paulo: Makron Books. 1995.
    - VELOSO, Paulo A. S. Complexidade de algoritmos: análise, projeto e métodos. Porto Alegre: Sagra Luzzatto, 2001.

Algoritmos e Estruturas de Dados III