

Teoria dos Grafos

Aula 4

Aula passada

- Representando grafos
- Matriz e lista

Aula de hoje

- Eplorando grafos
- Mecanismos genéricos
- Ideias sobre BFS, DFS

Busca em Grafos

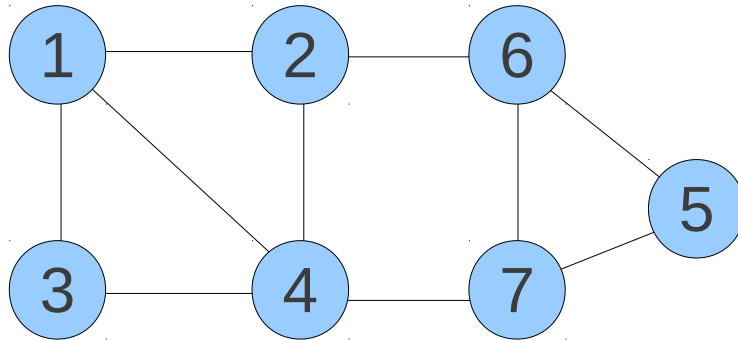
- Problema fundamental em grafos

Como *explorar* um grafo de forma sistemática?

- Muitas aplicações são abstraídas como problemas de busca
- Muitos algoritmos utilizam fundamentos similares

Busca em Grafos

- Como saber se existe caminho entre dois vértices?
 - de maneira eficiente



- **Idéia:** evitar explorar vértices já explorados
 - marcar os vértices!
 - vértice: *descoberto* ou *explorado*

Busca em Grafos

- Definir vértice inicial (origem ou raiz)
- Explorar e marcar vértices
 - *Descoberto*: vértice foi descoberto (visitado pela primeira vez)
 - *Explorado*: todas as arestas incidentes ao vértice foram exploradas e vizinhos descobertos
- Algoritmo genérico?

Algoritmo Genérico

- Passo inicial
 - desmarcar todos os vértices
 - selecionar origem e marcá-lo *descoberto*
- Passo geral (enquanto houver vértice descoberto)
 - Selecionar vértice descoberto, u
 - Considerar aresta não explorada, (u, v)
 - Se v não estiver marcado, marcar v como *descoberto*
 - Marcar u *explorado* quando não houver mais arestas incidentes a u a serem exploradas

Ordenação da Exploração

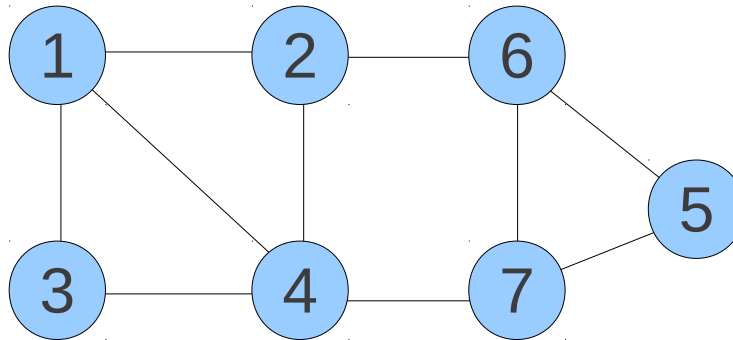
- Ordem de visita dos vértices e arestas?

Algoritmo genérico não estabelece ordem

- Qual é uma possível ordem?
 - Sistemática de exploração
- Duas abordagens
 - Explorar o vértice *descoberto* “mais antigo”
 - Explorar o vértice *descoberto* “mais recente”

Busca em Largura (BFS)

- Explorar vértices descobertos mais antigos primeiro

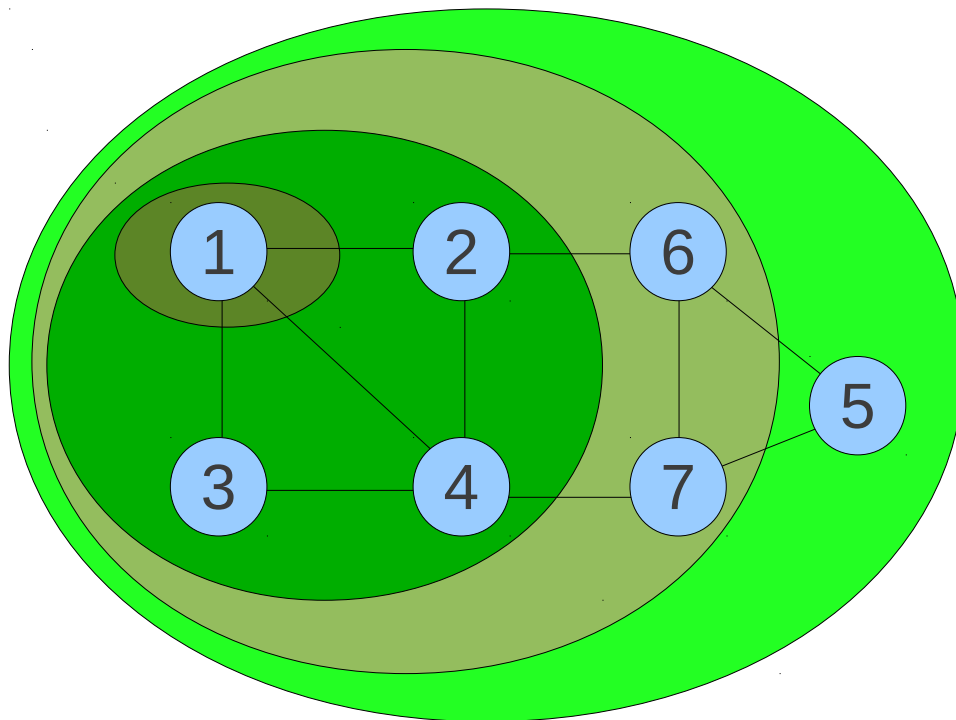


- Origem: vértice 1
- Em que ordem os vértices são *descobertos*?

Assumir arestas são exploradas em ordem crescente dos vértices adjacentes (matriz ou lista de adjacência)

Interpretação

- Onda é propagada à partir da raiz
- Onda expande em círculos, descobrindo vértices alcançáveis!

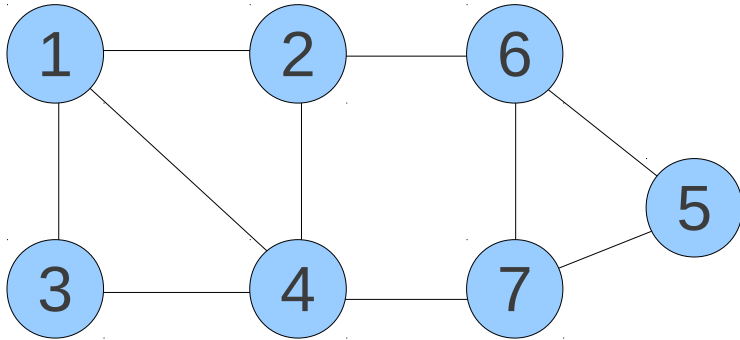


Busca em Largura!

Camadas

- L_i : conjunto de vértices pertencentes a camada $i=0, 1, 2, \dots$
- L_0 : vértice origem
- L_{i+1} : conjunto de vértices que **não fazem parte** de uma camada anterior e que **possuem uma aresta** com algum vértice da camada L_i

Camadas: Exemplo

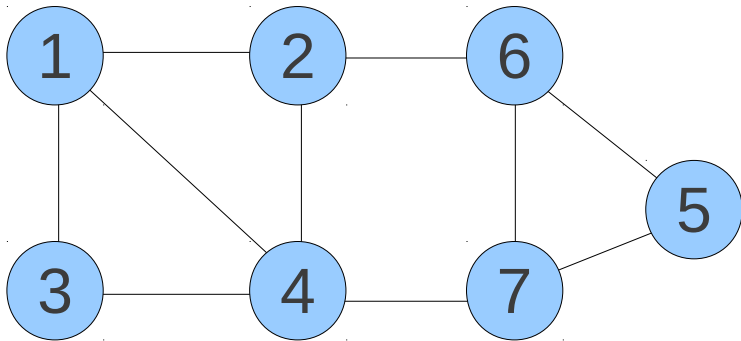


■ L_0 : vértice 2

■ L_i : ?

Distância

- Comprimento do **menor** caminho simples entre dois vértices
- Função $d(u,v)$, onde u e v são vértices
 - infinito quando não há caminho



■ Exemplo

- $d(1,2) = ?$
- $d(6, 3) = ?$
- $d(7, 1) = ?$

Camadas e Distância

- Qual é a relação entre eles?
- Vértices pertencentes a camada L_i têm distância i da origem!

**Busca em largura (BFS)
calcula distância!**

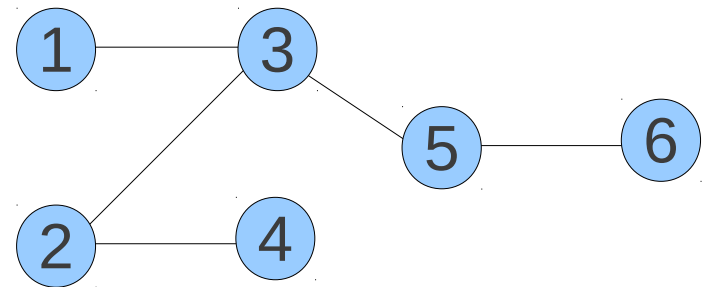
Grafo Acíclico

- Grafo acíclico é um grafo que não possui ciclos
 - lembrem do “ciclo”?

■ Exemplo:

- K_4 é acíclico?

É acíclico?

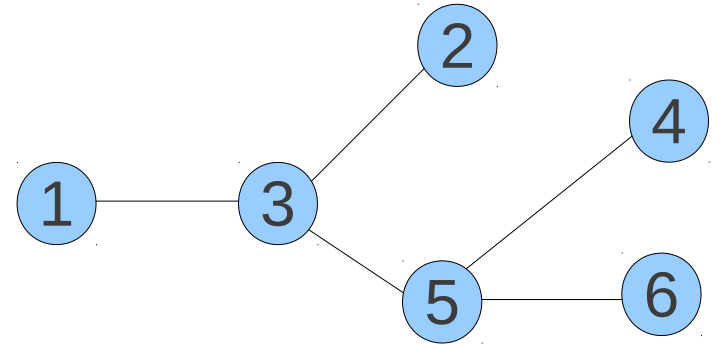


- Como descobrir se um grafo é acíclico?

Algoritmo eficiente!

Árvores

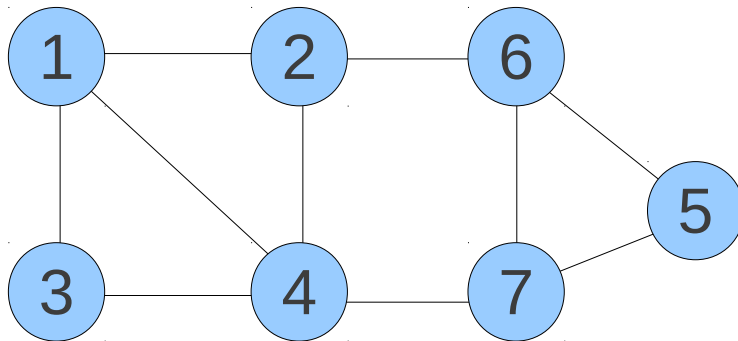
- Uma árvore é um grafo acíclico conexo
 - definição de árvore!



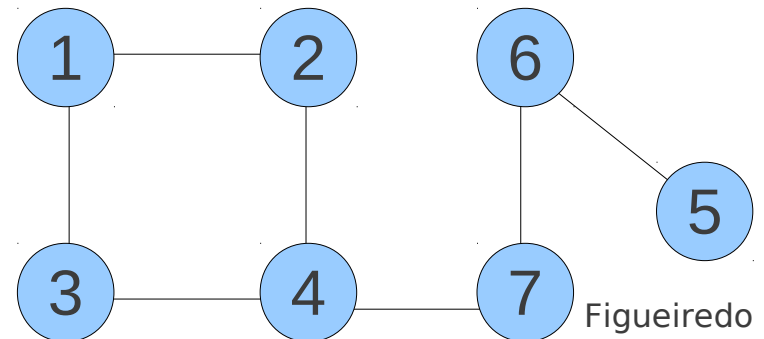
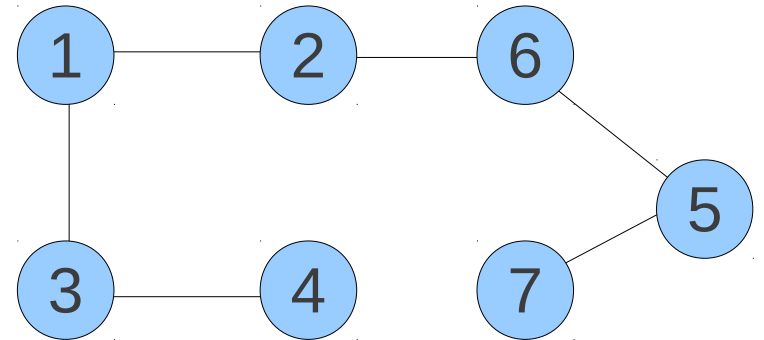
- Folha: vértice com grau 1
- Raiz: um determinado vértice
 - define orientação na árvore (pai, filhos, descendentes e acenstrais)
- Quantos caminhos (simples) distintos existe entre dois vértices quaisquer?
- Quantas arestas possui uma árvore com n vértices?

Árvore Geradora

- Subgrafo que contém **todos** os vértices de G e é uma **árvore**
 - em inglês, “spanning tree”
 - árvore que “alcança” todos os vértices



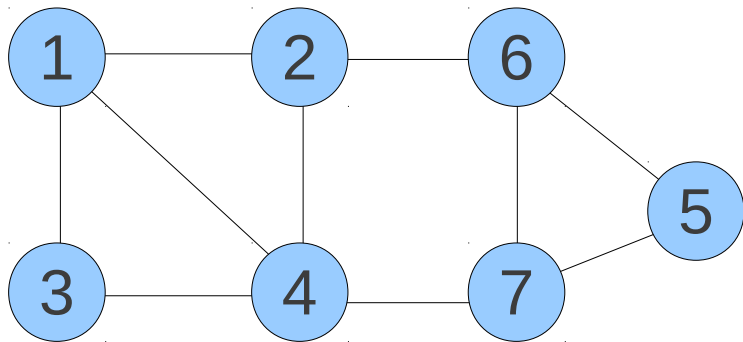
É árvore geradora?



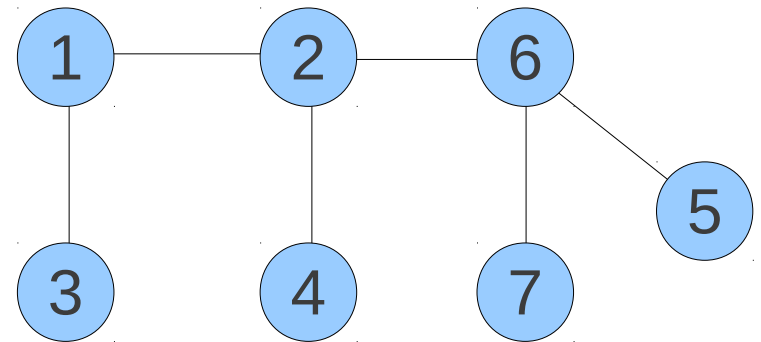
Árvore Geradora da BFS

- Árvore *induzida* pela busca em largura
 - Raiz: vértice de origem
 - Pai de v : nó que levou à descoberta de v

raiz: nó 6



Árvore geradora

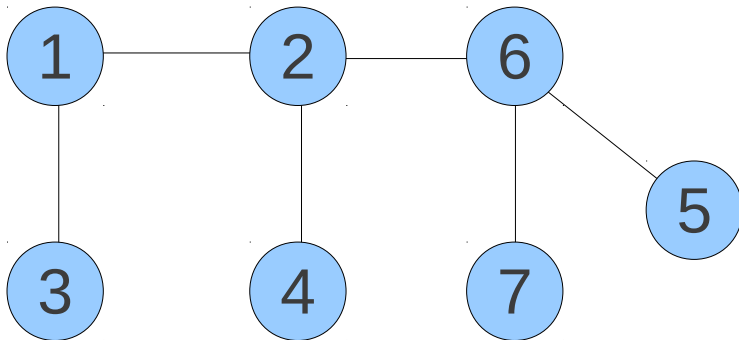


- Ordem da busca *define* árvore
- L_i = nível i da árvore

Menor Caminho

- Árvore geradora define menor caminho
- Dado vértice v (raiz) e outro vértice w qq.
 - menor caminho definido pela sequência de pais de w até a raiz

Árvore geradora
(raiz 6)



- menor caminho entre 3 e 6?
- menor caminho entre 3 e 7?
- **Cuidado!** Árvore define menor caminho para raiz!