

# Grafos - Aula 4

## Roteiro

- Percorrendo grafos
- Algoritmo genérico
- BFS
- Camadas
- Árvore geradora
- Caminhos mínimos

# Percorrendo Grafos

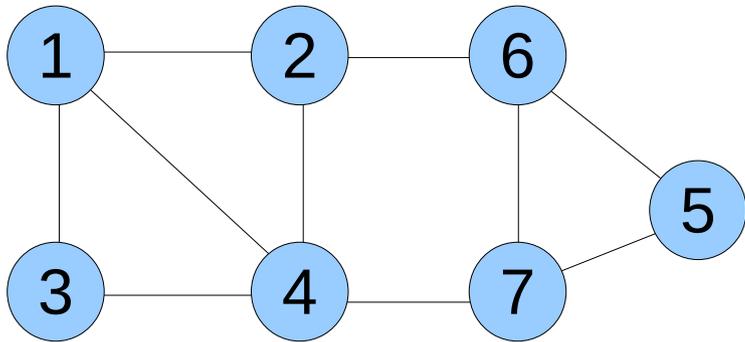
## Como *explorar* um grafo de forma sistemática?

*Explorar* = “passear pelo grafo”, “percorrer o grafo por suas arestas”

- Explorar para encontrar → buscar
- Muitas aplicações recaem em problemas de busca em grafos
- Muitos algoritmos utilizam fundamentos similares aos algoritmos de busca

# Exemplo

- Determinar se existe caminho entre dois vértices?



$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

**Algoritmo (eficiente)?**

# Algoritmo de Busca

- **Ideia:** evitar revisitar vértices já explorados
  - marcar os vértices
- **Marcações:** *desconhecido*, *descoberto* ou *explorado*
  - *Desconhecido:* vértice ainda não foi descoberto pelo algoritmo
  - *Descoberto:* vértice foi encontrado (visitado pelo algoritmo)
  - *Explorado:* todos vizinhos do vértice foram descobertos (todas as arestas incidentes ao vértice foram analisadas)

# Algoritmo Genérico

- Passo inicial
  - marcar todos os vértices como *desconhecidos*
  - selecionar origem e marcá-la *descoberto*
- Passo geral
  - Enquanto houver vértice *descoberto*
    - Selecionar algum vértice *descoberto*,  $u$
    - Se  $u$  possuir vizinho desconhecido,  $v$ 
      - marcar  $v$  como *descoberto*
    - Senão, marcar  $u$  como *explorado*

**O que teremos ao final?**

- vértices desconhecidos e vértices explorados

# Ordenação da Exploração

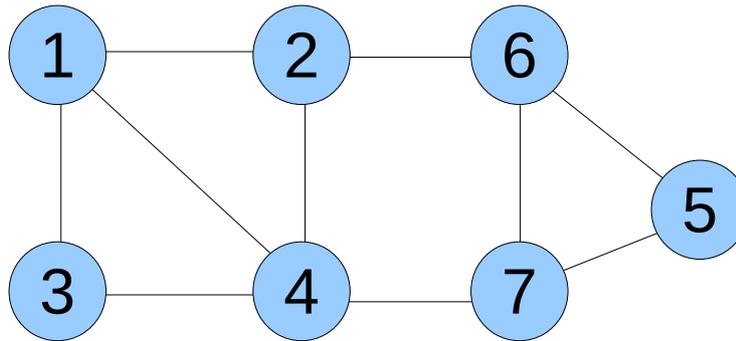
- Qual vértice  $u$  descoberto deve ser escolhido para ser analisado?

**Algoritmo genérico não estabelece ordem**

- Ordenação define uma sistemática de exploração
- Duas abordagens
  - $u$  é o vértice *descoberto* “mais antigo”
  - $u$  é o vértice *descoberto* “mais recente”

# Busca em Largura (BFS)

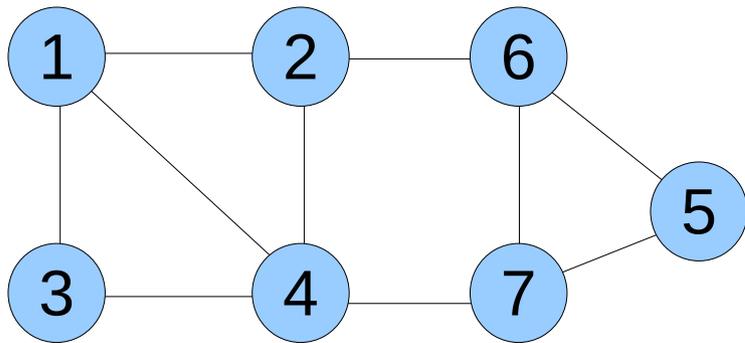
- Analisar vértices descobertos mais antigos primeiro



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

**Assumir que vizinhos são descobertos em ordem crescente de seus identificadores**

# Exemplo

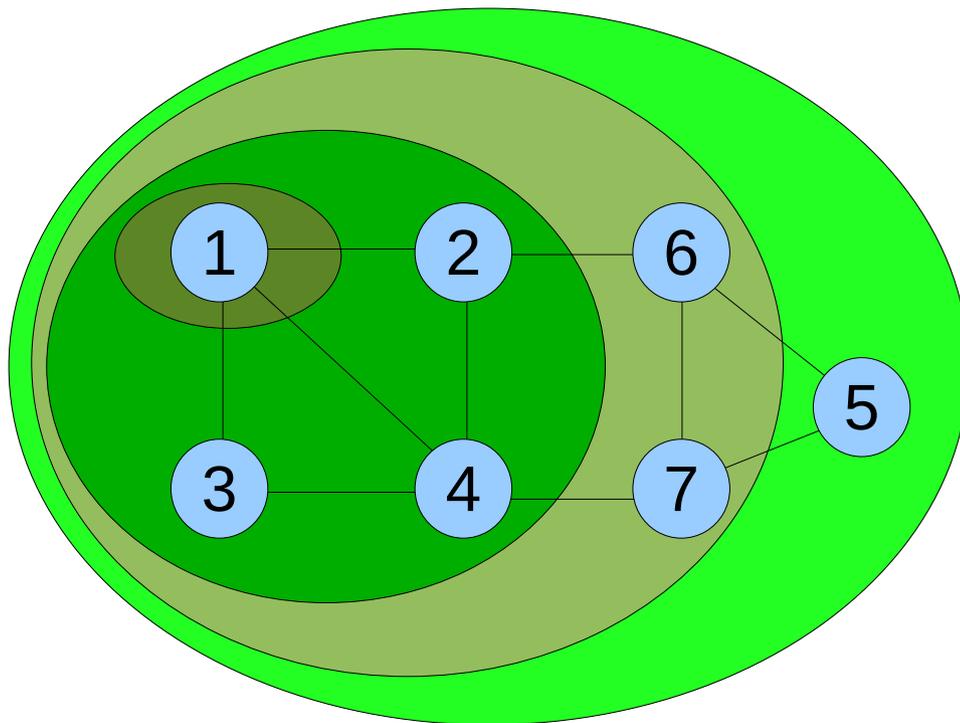


- Iniciar com vértice 1
- Analisar mais antigo primeiro
- Ordem que vértices são adicionados no conjunto descoberto
- 1,2,3,4,6,7,5

Passo	Descoberto	Explorado	Desconhecido
0	1	-	2,3,4,5,6,7
1	1,2	-	3,4,5,6,7
2	1,2,3	-	4,5,6,7
3	1,2,3,4	-	5,6,7
4	2,3,4	1	5,6,7
5	2,3,4,6	1	5,7
6	3,4,6	1,2	5,7
7	4,6	1,2,3	5,7
8	4,6,7	1,2,3	5
9	6,7	1,2,3,4	5
10	6,7,5	1,2,3,4	-
11	7,5	1,2,3,4,6	-
12	5	1,2,3,4,6,7	-
13	-	1,2,3,4,6,7,5	-

# Interpretação

- Onda é propagada à partir da raiz (origem)
- Onda expande em círculos, descobrindo vértices vizinhos dos vizinhos...

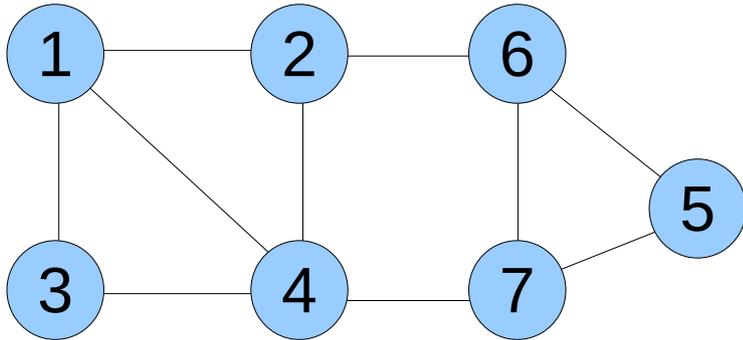


**Busca em Largura!**

# 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  $L_i$
- Conjuntos  $L_i$  dependem apenas do vértice de origem e do grafo

# Camadas: Exemplo



■  $L_0$  : vértice 2



$L_0$  : 2

■  $L_i$  : ?

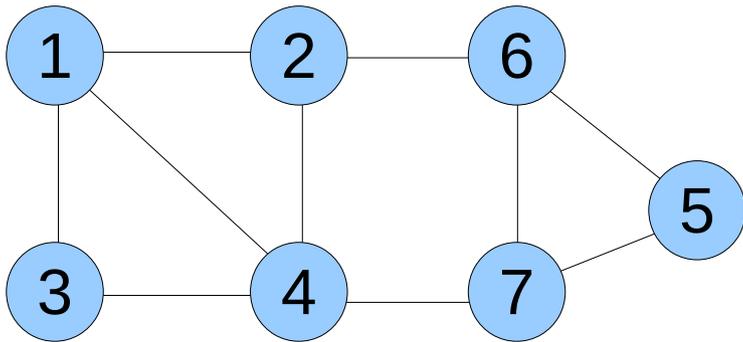
$L_1$  : 1, 4, 6

$L_2$  : 3, 5, 7

$L_3$  : vazia

# 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 camadas e distância?
- Vértices pertencentes a camada  $L_i$  têm distância  $i$  da origem
- Prova: por indução em  $i$ 
  - hipótese: vértices da camada  $L_i$  têm distância  $i$
  - indução: vértices da camada  $L_{i+1}$  têm distância  $i+1$
  - são vizinhos de vértices em  $L_i$  que possuem distância  $i$ , e não são vizinhos de vértices de camadas inferiores a  $i$

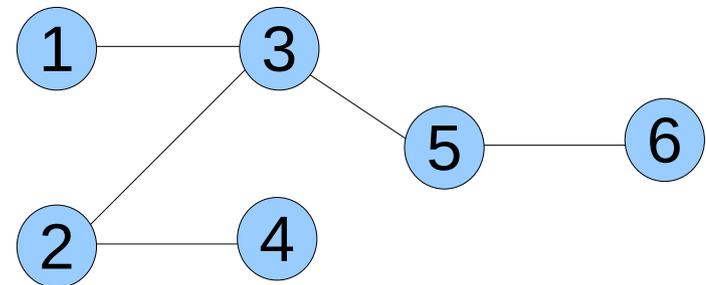
# Grafo Acíclico

- Grafo acíclico é um grafo que não possui ciclos
  - lembram do ciclo? Caminho simples que inicia e termina no mesmo vértice

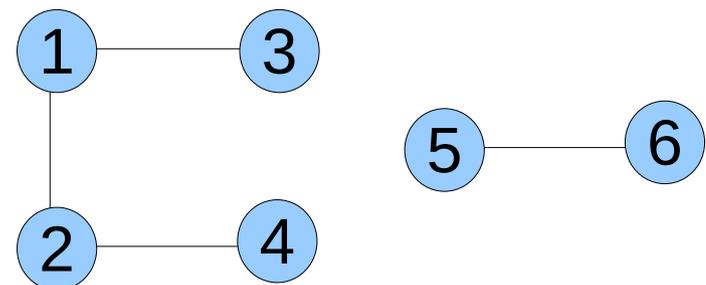
- Exemplo:

- $K_4$  é acíclico?

É acíclico?

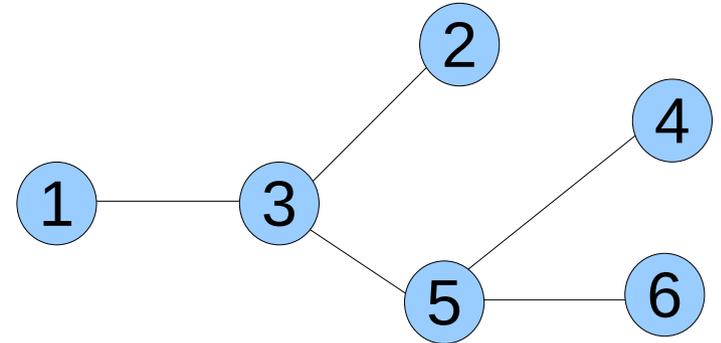


É acíclico?



# Árvores

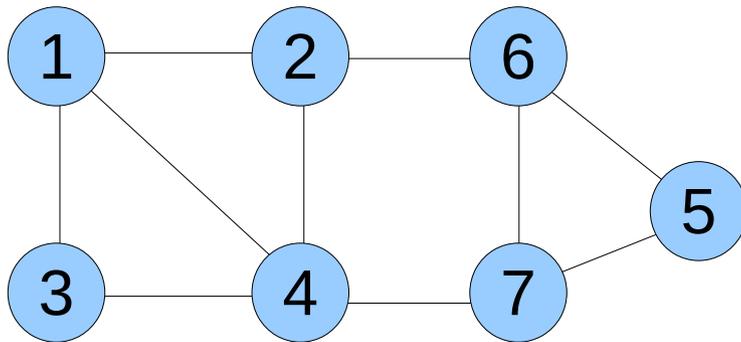
- Uma árvore é um grafo acíclico conexo
  - definição de árvore!



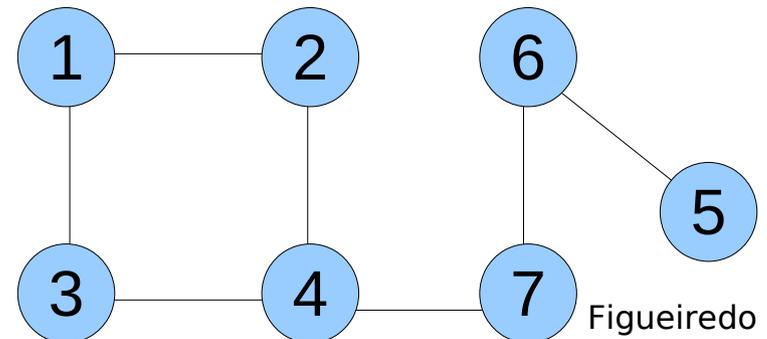
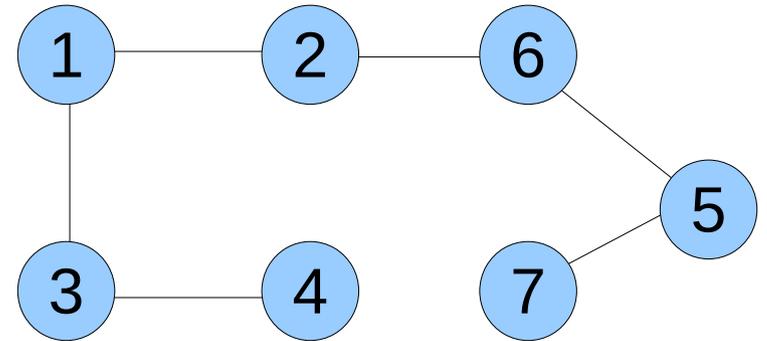
- Folha: vértice com grau 1
- Raiz: um determinado vértice
  - define orientação na árvore (pai, filhos, descendentes e acenstrais)
- Quantos caminhos (simples) existem entre dois vértices de uma árvore?
- Quantas arestas possui uma árvore com  $n$  vértices?

# Árvore Geradora

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



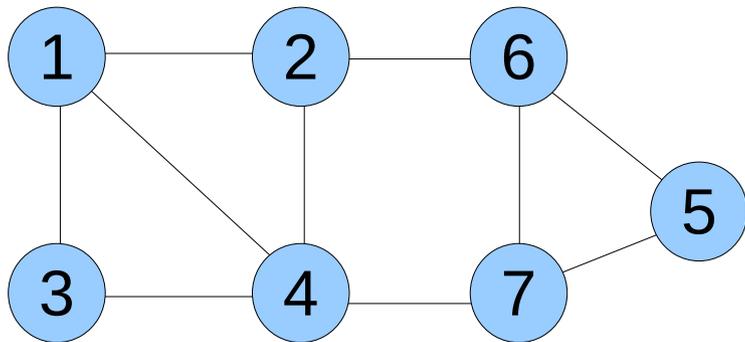
É árvore geradora?



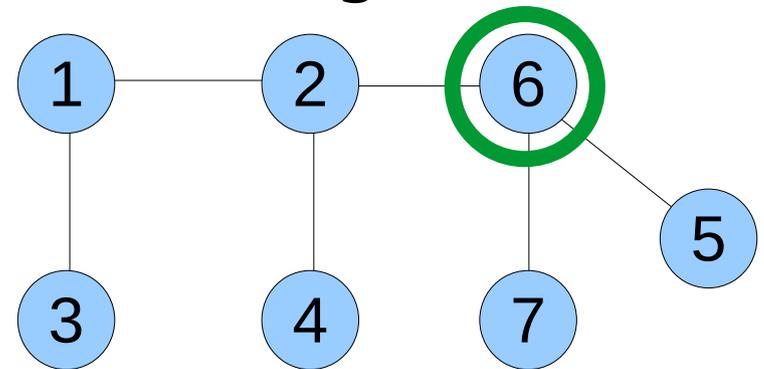
# Árvore Geradora da BFS

- *Árvore induzida* pela busca em largura
  - Raiz: vértice de origem
  - Pai de  $v$ : vértice que descobriu  $v$

raiz: nó 6



Árvore geradora

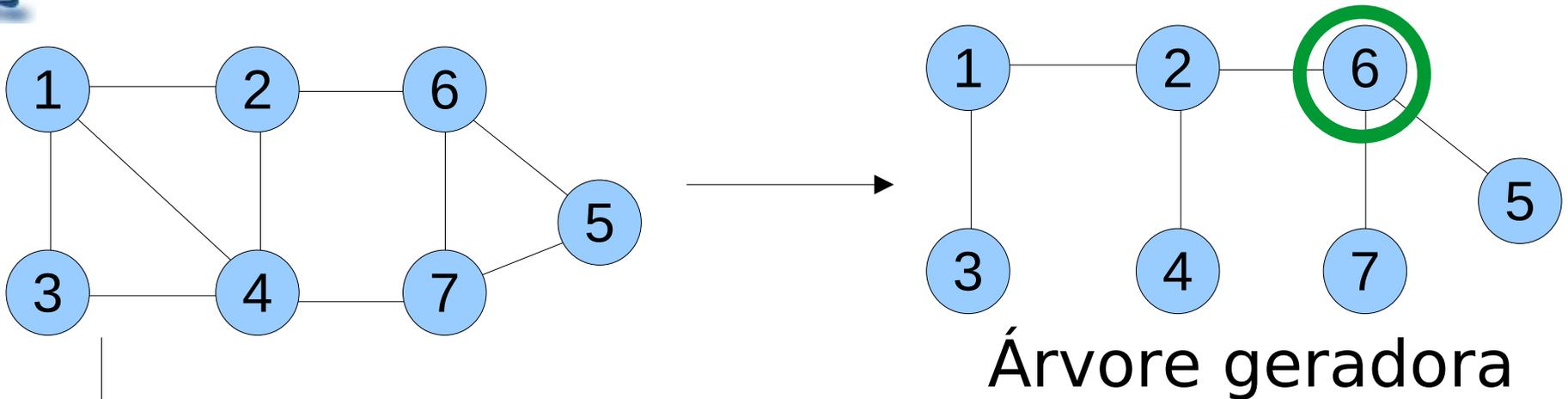


- Ordem da busca *define* árvore
  - vértice que descobriu um desconhecido

# Camadas e Árvore Geradora



- Qual é a relação entre camadas  $L_i$  e a árvore geradora?



$L_0 : 6$

$L_1 : 2, 5, 7$

$L_2 : 1, 4$

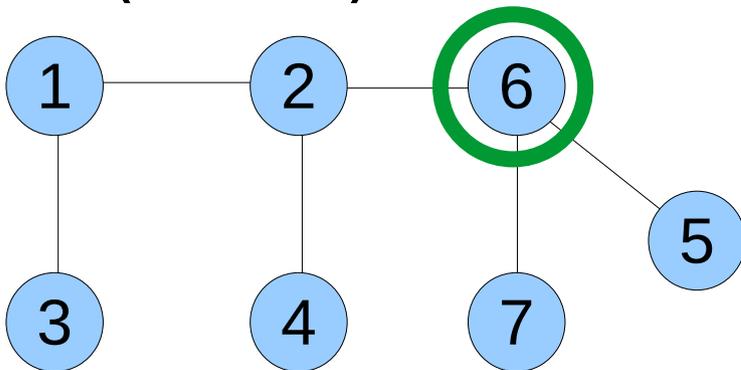
$L_3 : 3$

- Conjunto  $L_i =$  vértices no nível  $i$  da árvore
  - vértice está no nível  $i$  se o pai dele está no nível  $i-1$
- BFS calcula distâncias!

# 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!

# Poderosa BFS

- Determina se existe caminho entre dois vértices
- Calcula distância entre dois vértices
  - nível do vértice na árvore geradora
- Determina (um) menor caminho entre dois vértices
  - sequência de vértices na árvore geradora
- Calcular para dois vértices tem mesmo custo (pior caso) que calcular entre um

**Algoritmo na próxima aula!**