

# Grafos – Aula 6

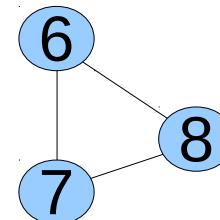
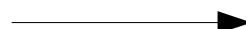
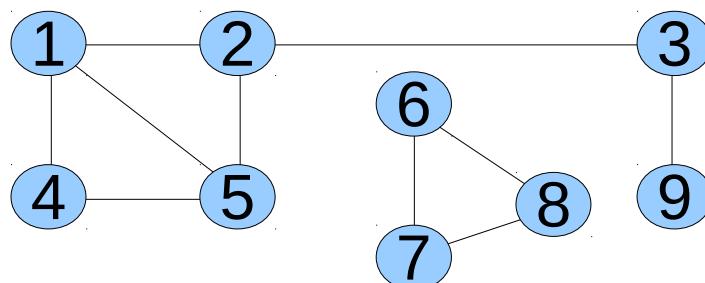
## Roteiro

- Componentes conexas
- Grafos direcionados
- Busca em grafos direcionados
- Grafos fortemente conexos

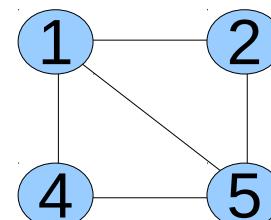
# Componentes Conexas

- Maiores subgrafos “conectados” de um grafo
- Subgrafos maximais de  $G$  que sejam conexos
  - *maximal*: maior subconjunto que possui a propriedade, no caso subgrafo conexo

- Exemplo:



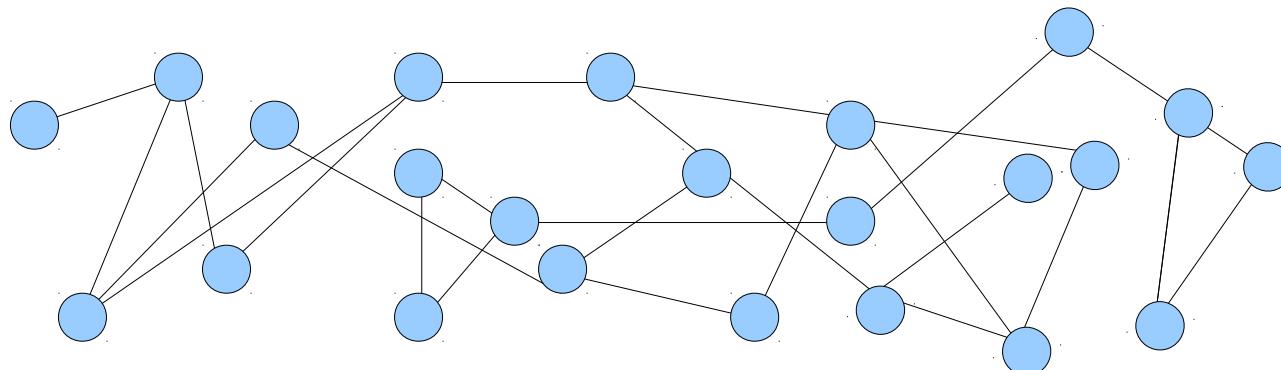
Componente  
conexa?



Componente  
conexa?

# Componentes Conexas

- **Problema:** Determinar o número de componentes conexas de um grafo
  - e tamanho de cada componente



**Algoritmo eficiente para  
resolver este problema?**

# Componentes Conexas

## Usar Busca em Grafos (BFS ou DFS)

- Desmarcar todos os vértices
- Escolher vértice s qualquer
- Realizar BFS(s)
- Vértices marcados determinam uma CC
- Escolher vértice s qualquer não marcado
- Realizar BFS(s)
- Vértices marcados determinam outra CC
- ...

# Componentes Conexas

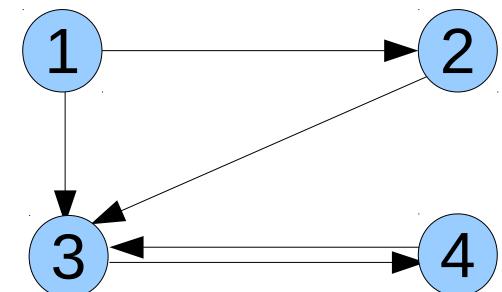
## Complexidade?

- Maior número de CC de um grafo?
  - $n$  = número de vértices
- Custo para percorrer cada CC?
  - tamanho da CC (em arestas e vértices)
- Usar marcações diferentes para cada CC
  - permite identificar as diferentes CC ao final
- Escolher vértice não marcado de forma eficiente
  - lista de vértices não marcados, atualizada

**$O(m + n)$**

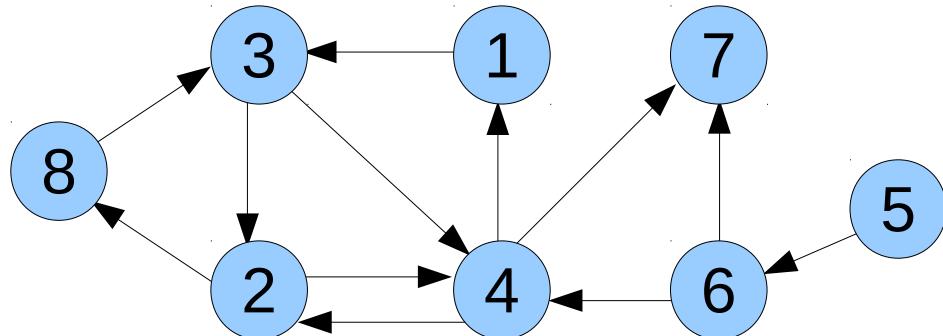
# Grafo Direcionado (Dígrafo)

- Relacionamento não é simétrico!
  - A estar relacionado com B, **não** implica B estar relacionado com A
  - Exemplo de relacionamento assimétrico?
- Abstração: arestas têm “direção”
  - par de vértices é ordenado
  - aresta aponta na ordem definida pelo par
- Exemplo:  $G = (V, E)$ 
  - $V = \{1, 2, 3, 4\}$
  - $E = \{(1,2), (1,3), (2,3), (3,4), (4,3)\}$



# Grau

- Grau de entrada: número de arestas que “entram” em  $v$ :  $|\{(*, v)\}|$
- Grau de saída: número de arestas que “saem” de  $v$ :  $|\{(v, *)\}|$
- Exemplo:  $G = (V, E)$

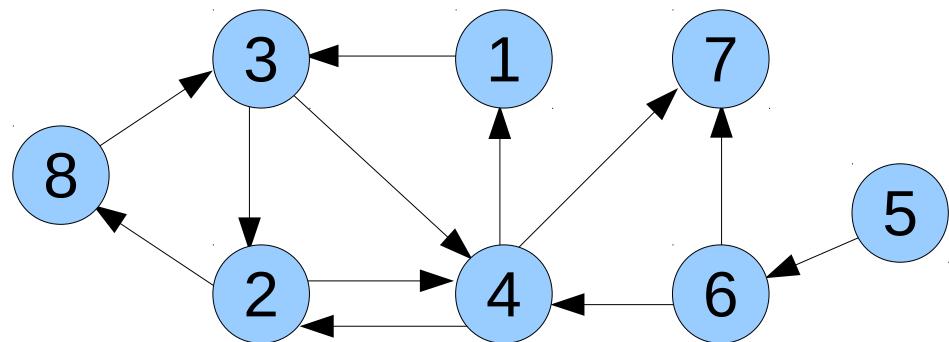


- $g_e(3) = ?$
- $g_s(3) = ?$
- $g_e(4) = ?$
- $g_s(7) = ?$

- Número máximo de arestas em  $G$ ?  $\leftarrow n(n-1)$
- Relação entre  $g_e(v)$  e  $g_s(v)$ ?  $\leftarrow$  Nenhuma

# Caminho, Ciclo, Distância

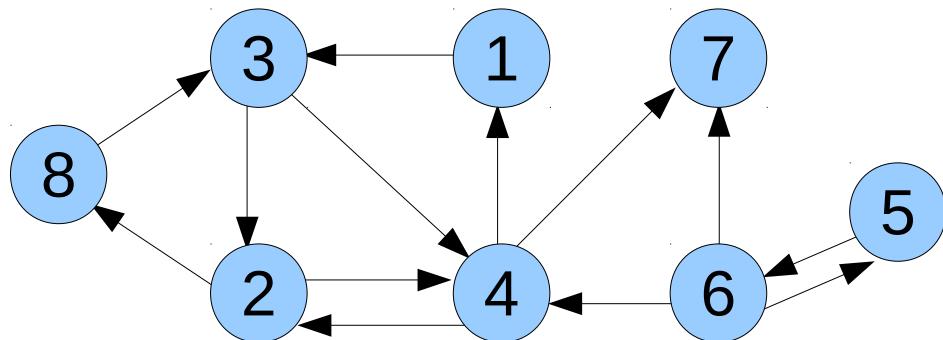
- Mesma definição de antes!
- Respeitando o direcionamento das areias
- Exemplo:  $G = (V, E)$



- Existe caminho de 5 para 2?
- Existe caminho de 8 para 6?
- Ciclo que contém 1?
- $d(1,8) = ?$
- $d(8,1) = ?$

# Fortemente Conexo

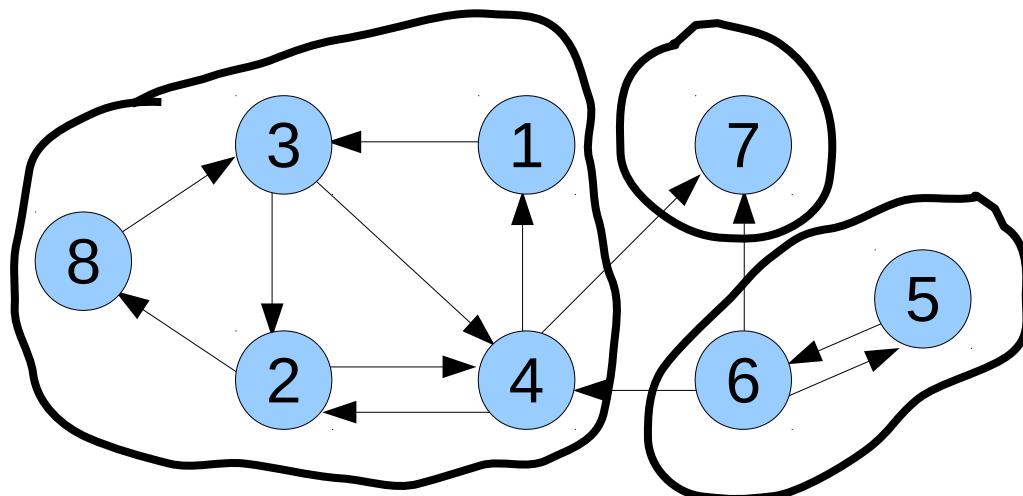
- Análogo a conexo (no caso não direcionado)
- Existe caminho entre qualquer par de vértices
  - repare que caminho de  $u$  para  $v$ , **não** implica caminho de  $v$  para  $u$
- Exemplo:  $G = (V, E)$



- É fortemente conexo?

# Componentes Fortemente Conexas

- Análogo a componentes conexas
- Subgrafos maximais de  $G$  que são fortemente conexos
- Exemplo:  $G = (V, E)$



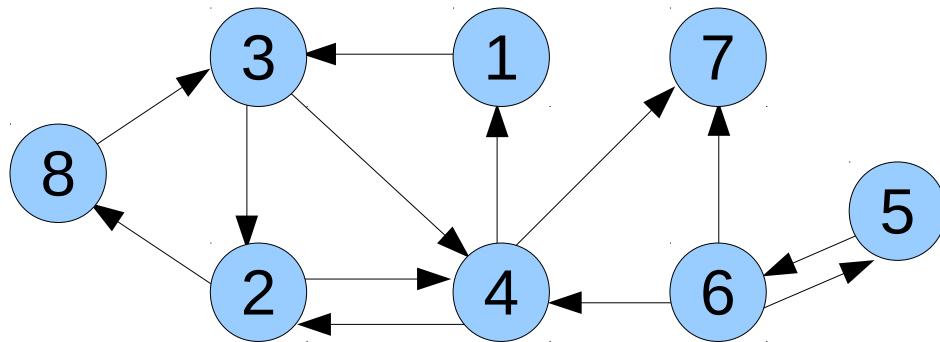
- Componentes fortemente conexas?

# Busca em Grafos Direcionados

**Mesma idéia!**

- Busca precisa respeitar direcionamento das arestas
  - $(u, v)$  não é igual a  $(v, u)$
- BFS e DFS idênticas (respeitando direcionamento das arestas)
- Mesmos algoritmos, mesma complexidade

# Exemplo



- BFS a partir do vértice 1
  - Ordem de descobrimento: 1,3,2,4,8,7
- DFS a partir do vértice 8
  - Ordem de descobrimento: 8,3,2,4,1,7

# Busca em Grafos Direcionados

- Escolher vértice  $s$  (grafo direcionado)
- Executar BFS à partir de  $s$
- Qual significado dos vértices marcados?

**Vértices que  $s$  alcança!**

- Tais vértices alcançam  $s$ ?
  - ou seja, existe caminho de volta a  $s$ ?

**Não necessariamente!**

# Busca em Grafos Direcionados

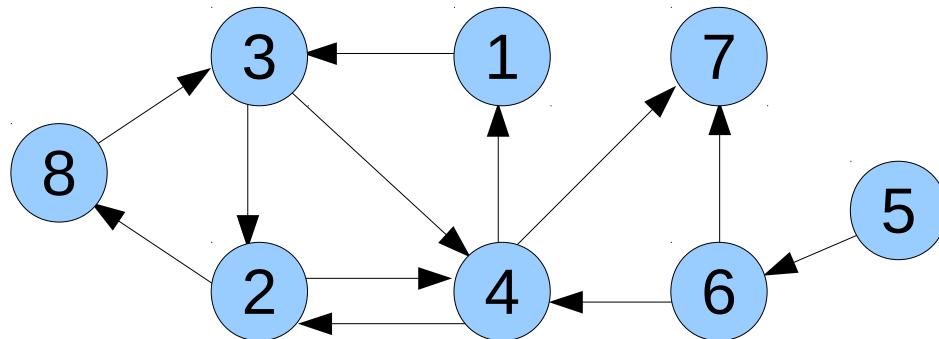
- Como descobrir quais vértices alcançam  $s$ ?
- Solução pouco eficiente
  - Para cada vértice do grafo, executar BFS e verificar se  $s$  é marcado

**Idéias melhores?**

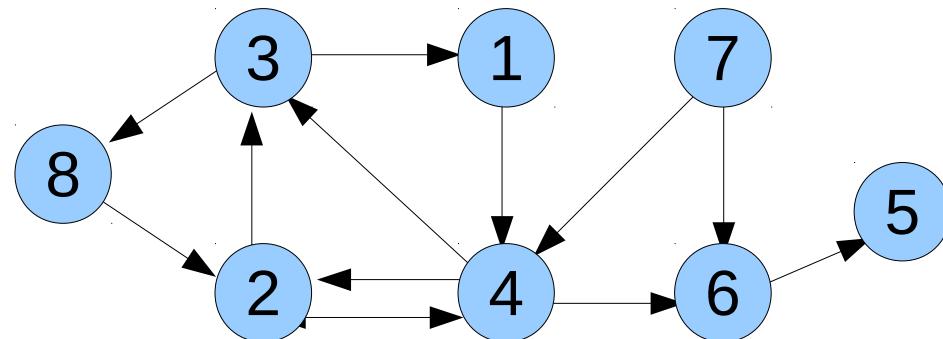
- Inverter a direção das arestas!
  - executar BFS no novo grafo à partir de  $s$
  - vértices que  $s$  alcançou, alcançam  $s$  no grafo original

# Exemplo

- Determinar vértices que alcançam 1?



- Grafo com arestas invertidas:  $G_{rev}$



- Execute BFS à partir de 1
- Vértices marcados alcançam 1 no grafo original
- Resultado: 1,4,2,3,6,8,5

# Fortemente Conexo

- **Problema:** Como determinar se um grafo direcionado é fortemente conexo?
- Como fizemos no caso não-direcionado?

**Idéias?**

- Mesmo princípio de antes
  - Se  $s$  chega aos vértices  $u$  e  $v$ , e os vértices  $u$  e  $v$  chegam a  $s$
  - Então  $u$  chega a  $v$ , via  $s$

# Fortemente Conexo

- **Problema:** Como determinar se um grafo direcionado é fortemente conexo?
- Escolher vértice  $s$  qualquer
- Executar BFS à partir de  $s$
- Construir  $G_{rev}$
- Executar BFS à partir de  $s$  em  $G_{rev}$
- Se todos os vértices foram marcados nas duas buscas, então  $G$  é fortemente conexo

**Complexidade?**

# Complexidade

- Escolher vértice  $s$  qualquer  $\xleftarrow{} O(1)$ , escolher vértice 1
- Executar BFS à partir de  $s$   $\xleftarrow{} O(m + n)$
- Construir  $G_{rev}$   $\xleftarrow{} O(m + n)$ , visitar todos os vértices e arestas
- Executar BFS à partir de  $s$  em  $G_{rev}$   $\xleftarrow{} O(m + n)$
- Se todos os vértices foram marcados nas duas buscas, então  $G$  é fortemente conexo  $\xleftarrow{} O(n)$ , percorrer as duas marcações

**O( $m + n$ )**

# Componentes Fortemente Conexas

- Como encontrar componentes fortemente conexas?
  - SCC = Strongly Connected Components
- **Ideia 0:** algoritmo anterior identifica a componente fortemente conexa que contém
  - vértices marcados duas vezes
  - iterar algoritmo sobre vértices não marcados
- Faz duas buscas por cada SCC
- **Ideia 1:** algoritmo de Tarjan
  - baseado na DFS, mantém histórico de conectividade
  - mais eficiente que ideia 0