



# Estruturas de Dados II

## Busca em Grafos

Prof. Leonardo C. R. Soares - *leonardo.soares@ifsudestemg.edu.br*

Instituto Federal do Sudeste de Minas Gerais

16 de agosto de 2024

---

<sup>a</sup>Este material é fortemente baseado nas notas de aula do professor Marco Antonio Moreira de Carvalho - UFOP



## Busca em grafos

## Definição

**A busca em grafos** (ou percurso em grafos) é a examinação de vértices e arestas de um grafo.

O projeto de bons algoritmos para determinação de estruturas ou propriedades de grafos depende fortemente do domínio destas técnicas.

## Terminologia

- Uma aresta ou vértice ainda não examinados são marcados como **não explorados** ou **não visitados**;





# Busca em grafos

## Definição

A **busca em grafos** (ou **percurso em grafos**) é a examinação de vértices e arestas de um grafo.

O projeto de bons algoritmos para determinação de estruturas ou propriedades de grafos depende fortemente do domínio destas técnicas.

## Terminologia

- ▶ Uma aresta ou vértice ainda não examinados são marcados como **não explorados** ou **não visitados**;
- ▶ Inicialmente, todos os vértices e arestas são marcados como não explorados;



## Busca em grafos

## Definição

**A busca em grafos** (ou percurso em grafos) é a examinação de vértices e arestas de um grafo.

O projeto de bons algoritmos para determinação de estruturas ou propriedades de grafos depende fortemente do domínio destas técnicas.

## Terminologia

- Uma aresta ou vértice ainda não examinados são marcados como **não explorados** ou **não visitados**;
  - Inicialmente, todos os vértices e arestas são marcados como não explorados;
  - Após terem sido examinados, os mesmos são marcados como **explorados** ou **visitados**;



## Busca em grafos

## Definição

**A busca em grafos** (ou percurso em grafos) é a examinação de vértices e arestas de um grafo.

O projeto de bons algoritmos para determinação de estruturas ou propriedades de grafos depende fortemente do domínio destas técnicas.

## Terminologia

- Uma aresta ou vértice ainda não examinados são marcados como **não explorados** ou **não visitados**;
  - Inicialmente, todos os vértices e arestas são marcados como não explorados;
  - Após terem sido examinados, os mesmos são marcados como **explorados** ou **visitados**;
  - Ao final, todos os vértices e arestas são marcados como explorados (no caso de uma busca completa).



## Busca genérica

**Entrada:** Grafo  $G=(V, A)$

```

1 enquanto existir  $j \in V$  com uma aresta  $\{j, k\}$  não visitada faça
2   Escolha o vértice  $j$  e visite a aresta  $\{j, k\}$ ;
3   se  $j$  não é marcado então
4     | marque  $j$  como visitado;
5   fim
6   se  $k$  não é marcado então
7     | marque  $k$  como visitado;
8   fim
9 fim

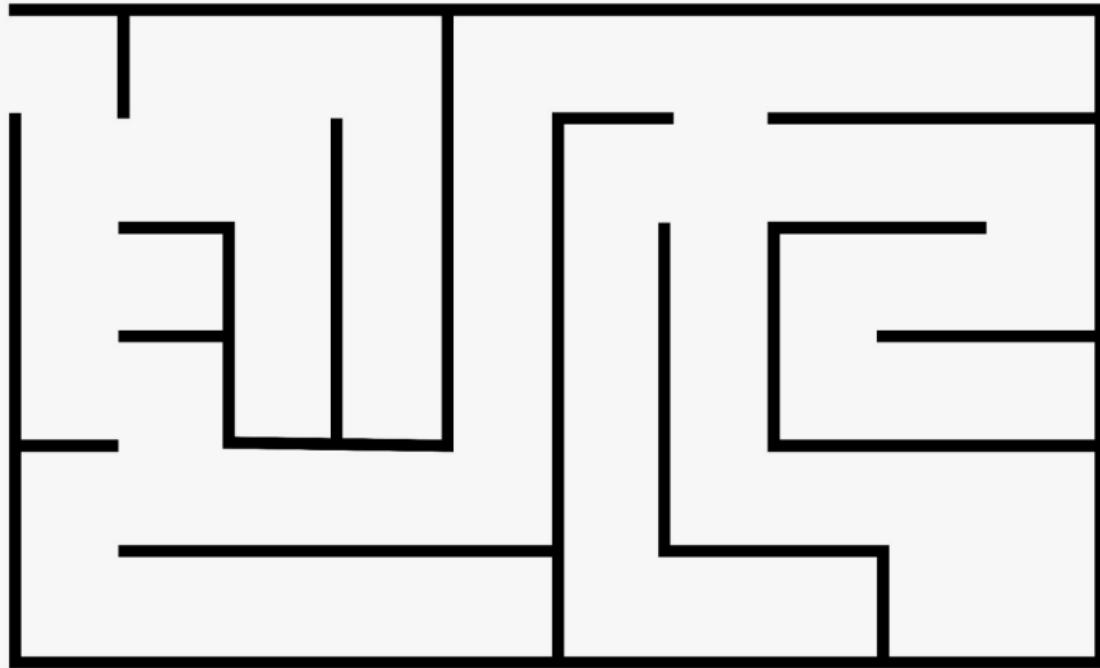
```



## Atravessando labirintos



# Como escapar?





# Como escapar?

## Método do século XIX - Algoritmo de Trémaux

O algoritmo de Trémaux requer que para encontrar a saída de um labirinto seja riscada uma linha no chão para marcar os caminhos percorridos:





# Como escapar?

## Método do século XIX - Algoritmo de Trémaux

O algoritmo de Trémaux requer que para encontrar a saída de um labirinto seja riscada uma linha no chão para marcar os caminhos percorridos:

- ▶ Um caminho pode ser **não visitado, visitado uma vez ou visitado duas vezes**;
- ▶ A cada escolha de direção, é riscado uma linha no chão indicando o caminho.



## Como escapar?

## Método do século XIX - Algoritmo de Trémaux

- Inicialmente, selecione uma direção aleatória;





# Como escapar?

## Método do século XIX - Algoritmo de Trémaux

- ▶ Inicialmente, selecione uma direção aleatória;
- ▶ Ao chegar em uma junção não visitada (nenhuma linha), escolha uma direção aleatória e risque o caminho;





# Como escapar?

## Método do século XIX - Algoritmo de Trémaux

- ▶ Inicialmente, selecione uma direção aleatória;
- ▶ Ao chegar em uma junção não visitada (nenhuma linha), escolha uma direção aleatória e risque o caminho;
- ▶ Ao chegar em uma junção já marcada, vire-se e caminhe de volta, marcando o caminho pela segunda vez;





# Como escapar?

## Método do século XIX - Algoritmo de Trémaux

- ▶ Inicialmente, selecione uma direção aleatória;
- ▶ Ao chegar em uma junção não visitada (nenhuma linha), escolha uma direção aleatória e risque o caminho;
- ▶ Ao chegar em uma junção já marcada, vire-se e caminhe de volta, marcando o caminho pela segunda vez;
- ▶ Se este não for o caso, escolha o caminho com menos linhas e marque-o novamente;





# Como escapar?

## Método do século XIX - Algoritmo de Trémaux

- ▶ Inicialmente, selecione uma direção aleatória;
- ▶ Ao chegar em uma junção não visitada (nenhuma linha), escolha uma direção aleatória e risque o caminho;
- ▶ Ao chegar em uma junção já marcada, vire-se e caminhe de volta, marcando o caminho pela segunda vez;
- ▶ Se este não for o caso, escolha o caminho com menos linhas e marque-o novamente;
- ▶ Ao chegar à saída, os caminhos marcados com apenas uma linha indicarão o caminho direto até o ponto inicial;





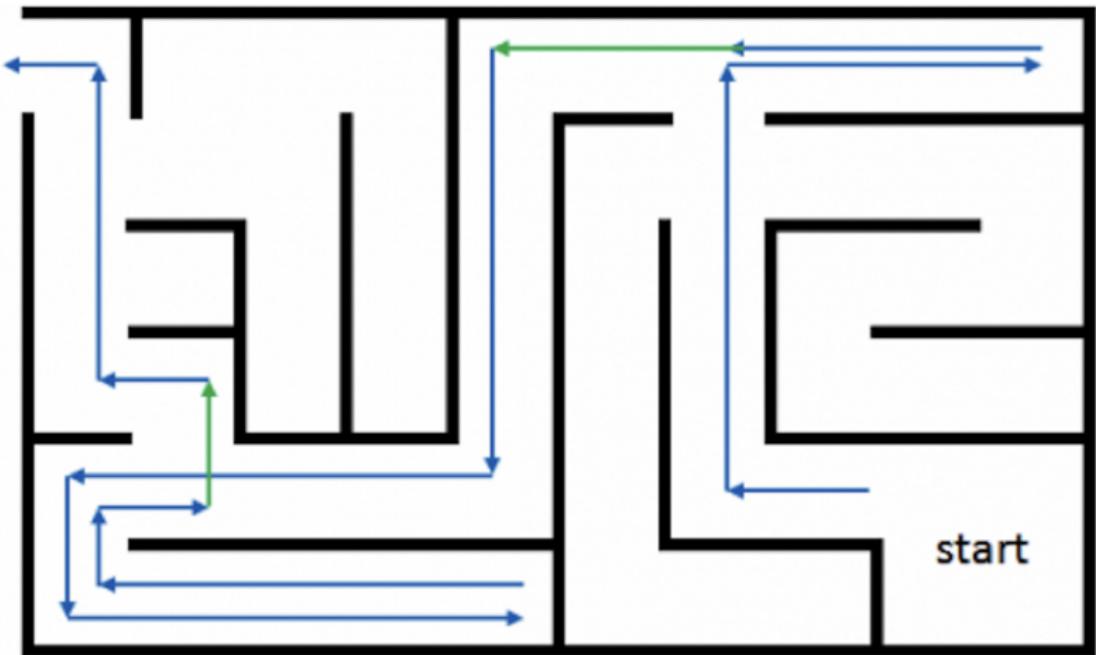
# Como escapar?

## Método do século XIX - Algoritmo de Trémaux

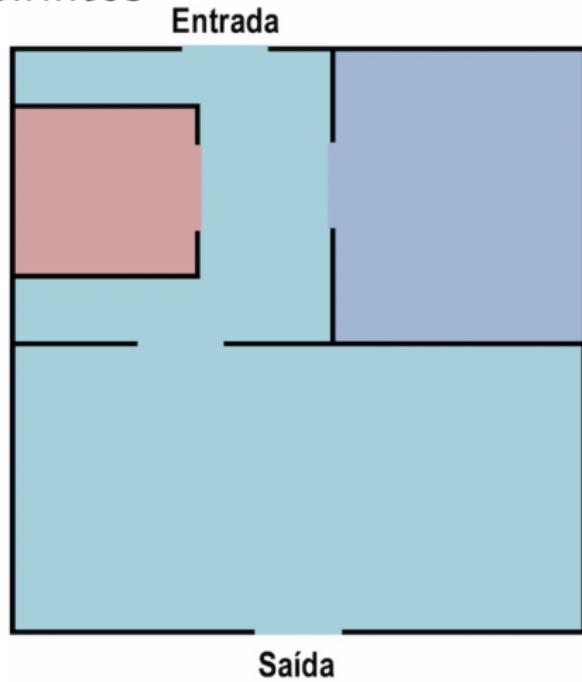
- ▶ Inicialmente, selecione uma direção aleatória;
- ▶ Ao chegar em uma junção não visitada (nenhuma linha), escolha uma direção aleatória e risque o caminho;
- ▶ Ao chegar em uma junção já marcada, vire-se e caminhe de volta, marcando o caminho pela segunda vez;
- ▶ Se este não for o caso, escolha o caminho com menos linhas e marque-o novamente;
- ▶ Ao chegar à saída, os caminhos marcados com apenas uma linha indicarão o caminho direto até o ponto inicial;
- ▶ Se não houver saída, você voltará ao ponto inicial, no qual todos os caminhos possuem duas linhas.



## Como escapar?



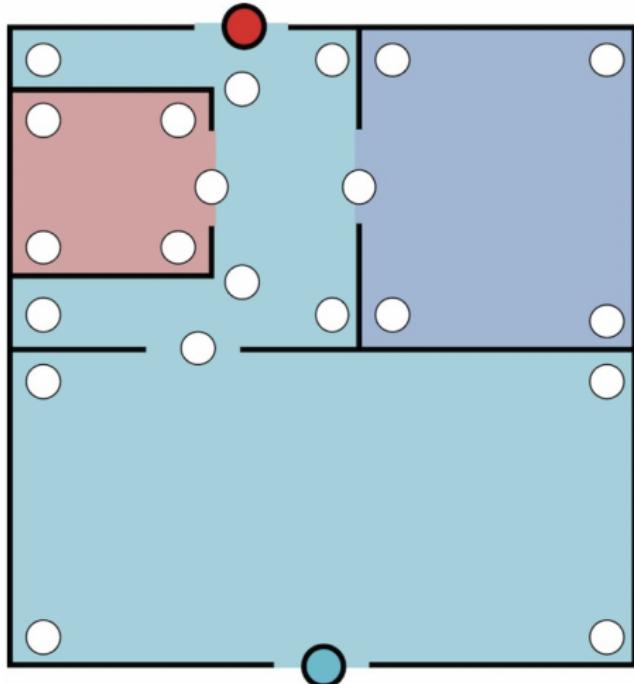
# Atravessando labirintos



Considera-se que a circulação neste labirinto é feita margeando alguma parede.

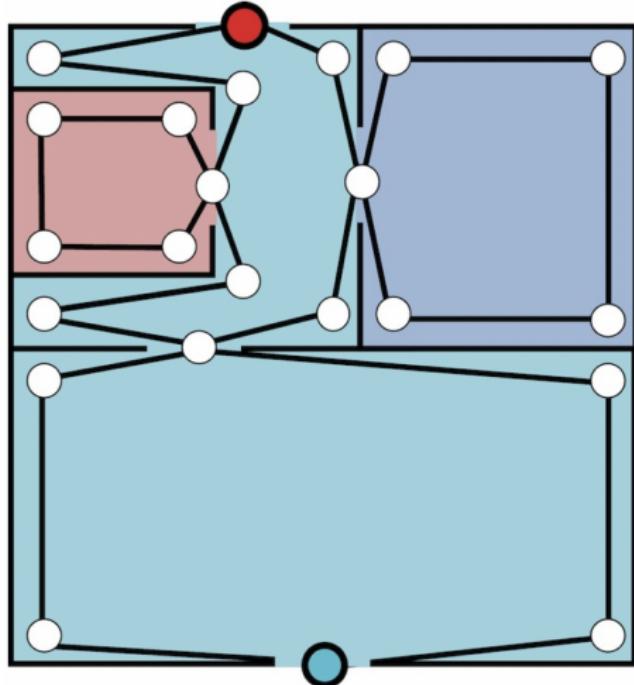


Atravessando labirintos



Pontos de entrada, saída e mudança de direção do labirinto.

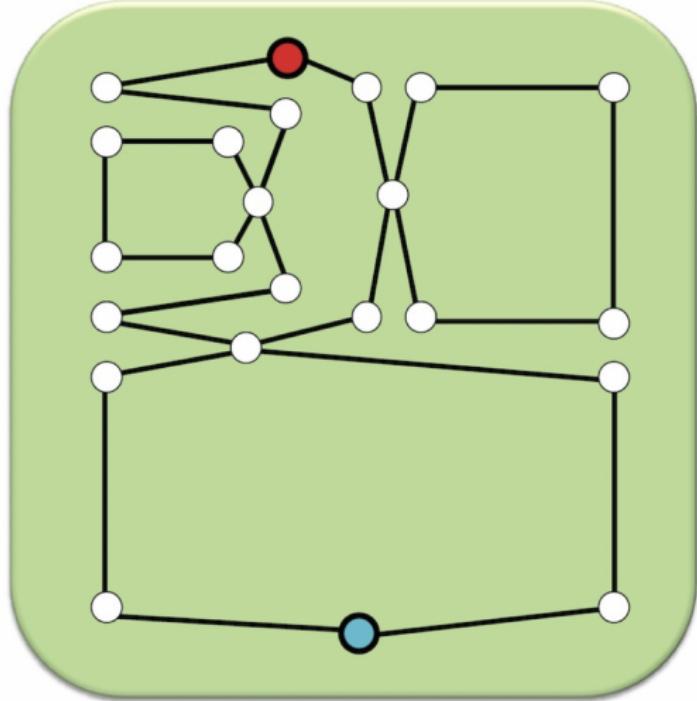
## Atravessando labirintos



Grafo no labirinto.



## Atravessando labirintos



Grafo no labirinto.



## Atravessando labirintos

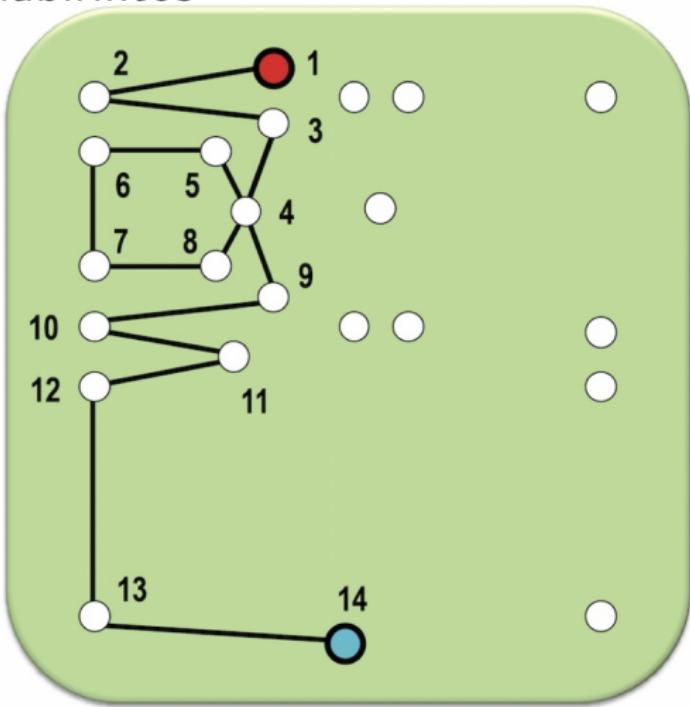
## Estratégia

Uma forma de encontrar a saída do labirinto sem nunca percorrer mais de uma vez o mesmo caminho consiste em realizar uma busca no grafo de modo a nunca repetir arestas, marcando-as.

A busca se inicia no vértice que representa a entrada e termina no vértice que representa a saída.



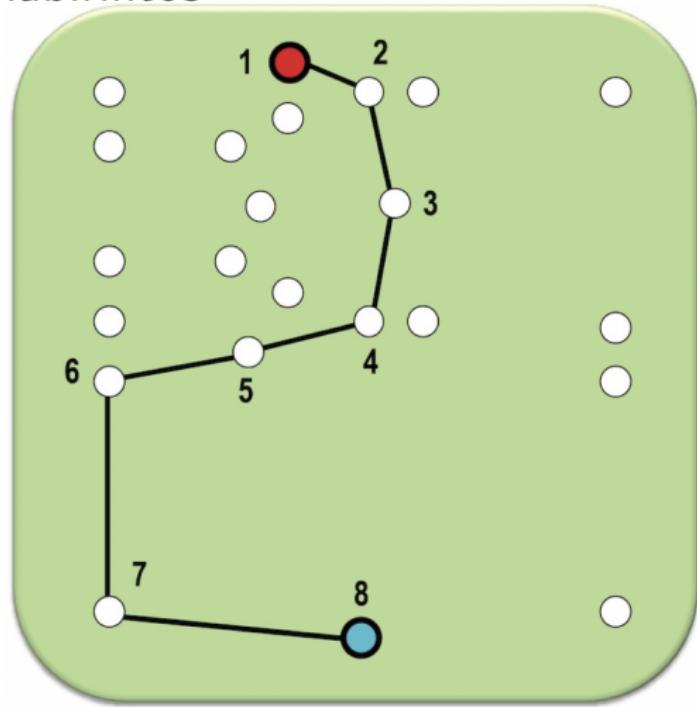
## Atravessando labirintos



Busca genérica no grafo de exemplo (1).



## Atravessando labirintos



Busca genérica no grafo de exemplo (2).





# Buscas em grafos

## Buscas em grafos

Dependendo do critério utilizado para escolha dos vértices e arestas a serem visitados, diferentes tipos de buscas são desenvolvidos a partir da busca genérica.





# Buscas em grafos

## Buscas em grafos

Dependendo do critério utilizado para escolha dos vértices e arestas a serem visitados, diferentes tipos de buscas são desenvolvidos a partir da busca genérica.

Basicamente, duas buscas completas em grafos são essenciais:

- ▶ Busca em profundidade (Depth-First Search, DFS);
- ▶ Busca em largura (Breadth-First Search, BFS).





# Busca em profundidade - DFS

## Características

A **busca em profundidade** visita todos os vértices de um grafo usando como critério os vizinhos do vértice visitado mais recentemente.





# Busca em profundidade - DFS

## Características

A **busca em profundidade** visita todos os vértices de um grafo usando como critério os vizinhos do vértice visitado mais recentemente.

**Característica principal:** Utiliza uma **pilha explícita** ou **recursividade** para guiar a busca.



## Busca em profundidade - DFS

**Entrada:** Grafo  $G=(V, A)$ , vértice inicial  $v$

- ```

1 Marque o vértice  $v$  como visitado;
2 enquanto existir  $w$  vizinho de  $v$  faça
3   se  $w$  é marcado como não visitado então
4     Visite a aresta  $\{v, w\}$ ;
5     BP( $G, w$ ); // chamada recursiva da função
6   fim
7   senão
8     se  $\{v, w\}$  não foi visitada ainda então
9       | Visite  $\{v, w\}$ ;
10    fim
11  fim
12 fim

```



## Busca em profundidade - DFS

## Classificação de arestas

Ao explorar um grafo  $G$  conexo usando a DFS, podemos categorizar as arestas:



## Busca em profundidade - DFS

## Classificação de arestas

Ao explorar um grafo  $G$  conexo usando a DFS, podemos categorizar as arestas:

- ▶ **Arestas de árvore:** Satisfazem ao primeiro se do algoritmo (linha 3), ou seja, levam à visitação de vértices ainda não visitados;
  - ▶ **Arestas de retorno:** Demais arestas. Formam ciclos pois levam a vértices já visitados (linha 8).





# Busca em profundidade - DFS

## Classificação de arestas

Ao explorar um grafo  $G$  conexo usando a DFS, podemos categorizar as arestas:

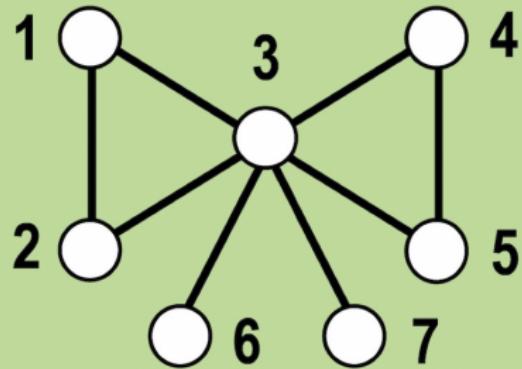
- ▶ **Arestas de árvore:** Satisfazem ao primeiro se do algoritmo (linha 3), ou seja, levam à visitação de vértices ainda não visitados;
- ▶ **Arestas de retorno:** Demais arestas. Formam ciclos pois levam a vértices já visitados (linha 8).

## Árvore de profundidade

A subárvore de  $G$  formada pelas vértices marcados e as *arestas de árvore* é chamada de **Árvore de Profundidade** de  $G$ .

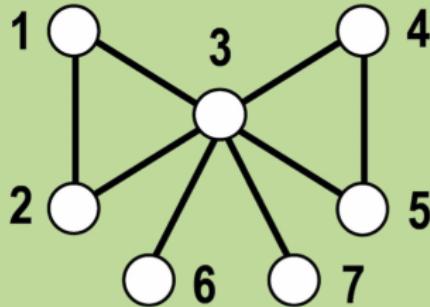


## DFS - Exemplo



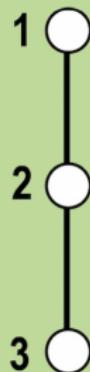
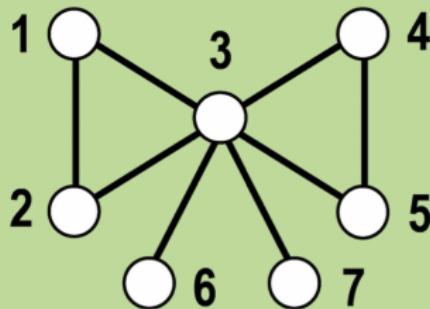
## Grafo de exemplo.

## DFS - Exemplo



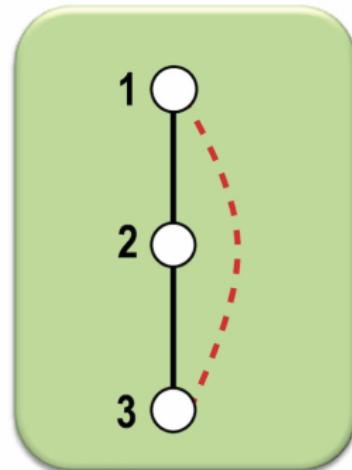
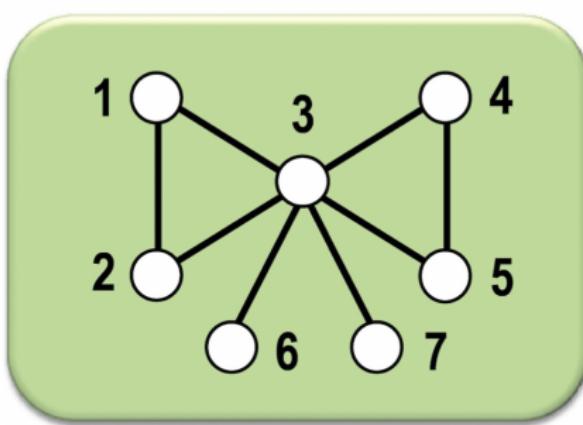
(1) Aresta  $\{1,2\}$ .

## DFS - Exemplo



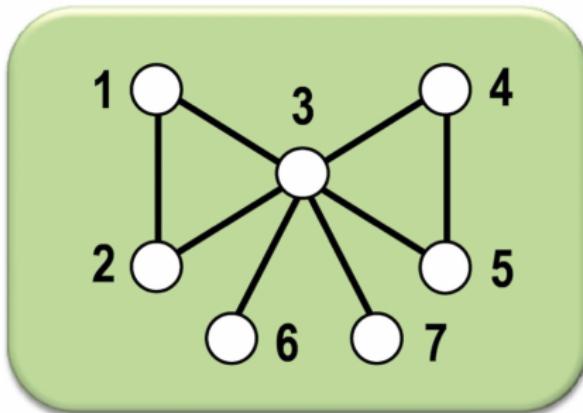
(2) Aresta  $\{2,3\}$ .

## DFS - Exemplo

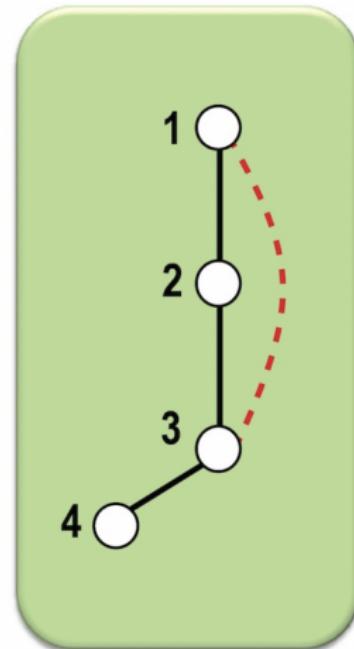


(3) Aresta  $\{3,1\}$ .

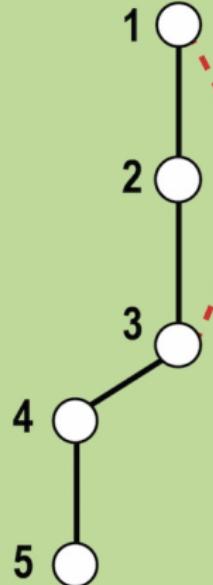
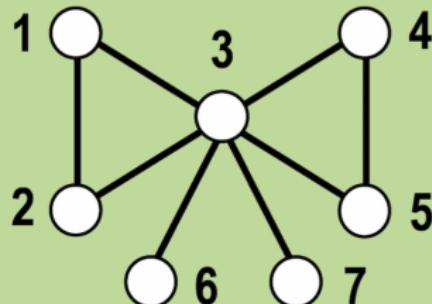
## DFS - Exemplo



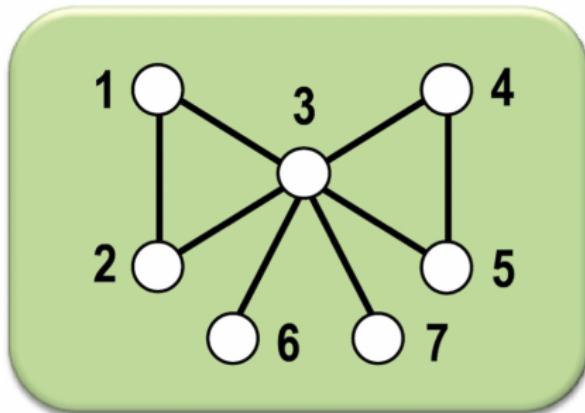
(4) Aresta {3,4}.



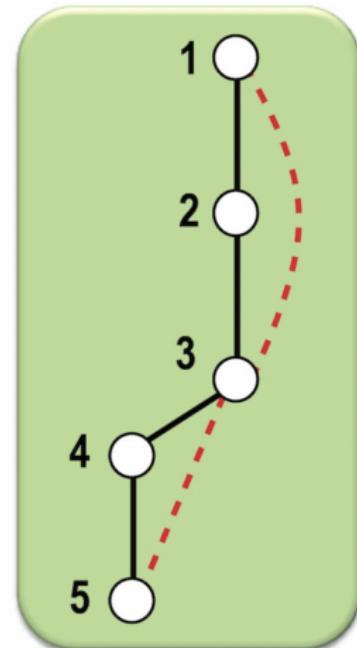
## DFS - Exemplo

(5) Aresta  $\{4,5\}$ .

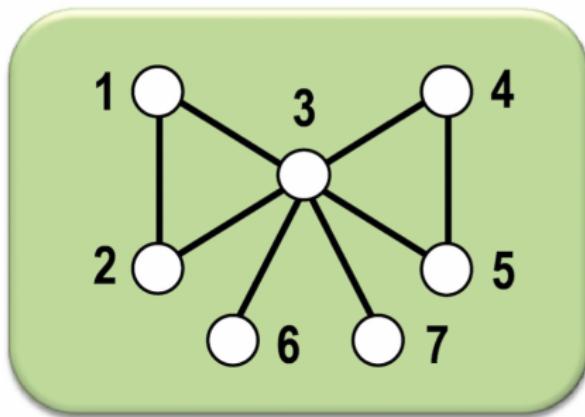
## DFS - Exemplo



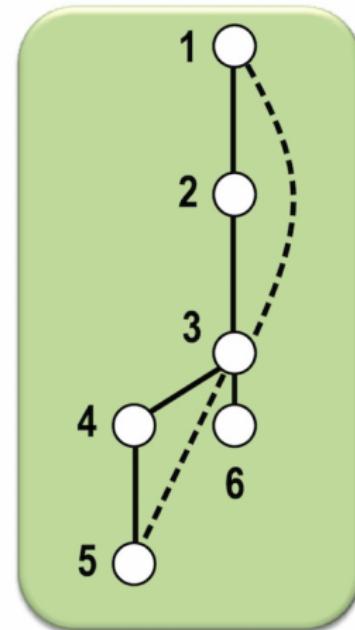
(6) Aresta  $\{5,3\}$ .



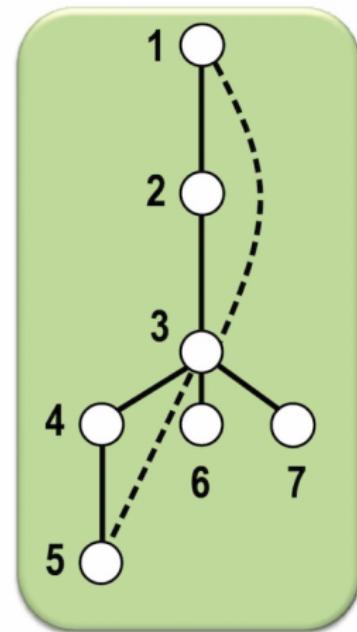
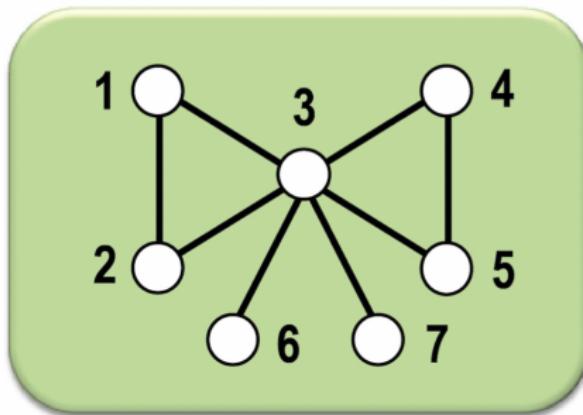
## DFS - Exemplo



(7) Aresta {3,6}.

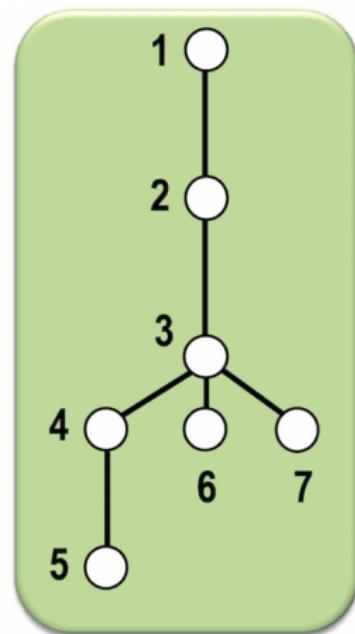
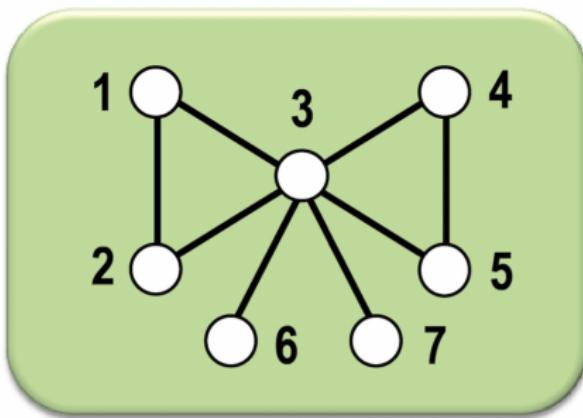


## DFS - Exemplo



(8) Aresta {3,7}.

## DFS - Exemplo



Grafo original e correspondente árvore de profundidade.



## DFS - Grafos direcionados

### Atenção:

A aplicação da DFS em grafos direcionados é essencialmente igual à aplicação em grafos não direcionados.

Mesmo o grafo direcionado sendo conexo, ou no caso de um GND desconexo, a DFS pode precisar ser chamada repetidas vezes enquanto houver vértices não visitados, retornando uma **floresta**.





# Busca em profundidade - Reinício

**Entrada:** Grafo  $G=(V, A)$

- 1 **enquanto** existir  $v \in V$  não visitado **faça**
- 2   |   BP( $G, v$ );
- 3 **fim**





# Busca em profundidade - DFS

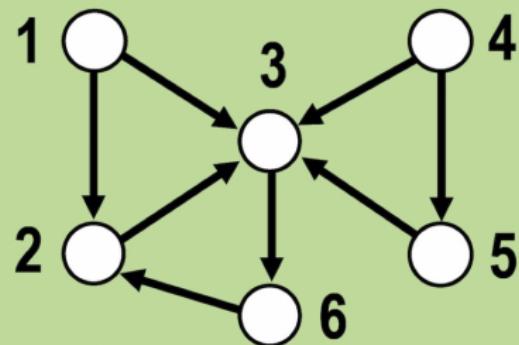
## Classificação de arcos

Ao explorar um grafo  $G$  direcionado usando a DFS, podemos categorizar os arcos. Sejam o vértice  $v$  a origem do arco e o vértice  $w$  o destino do mesmo:

- ▶ Arcos de avanço: Caso  $w$  seja descendente de  $v$  na floresta;
- ▶ Arcos de retorno: Caso  $v$  seja descendente de  $w$  na floresta;
- ▶ Arcos de cruzamento: Caso  $w$  não seja descendente de  $v$  e  $v$  não seja descendente de  $w$ .

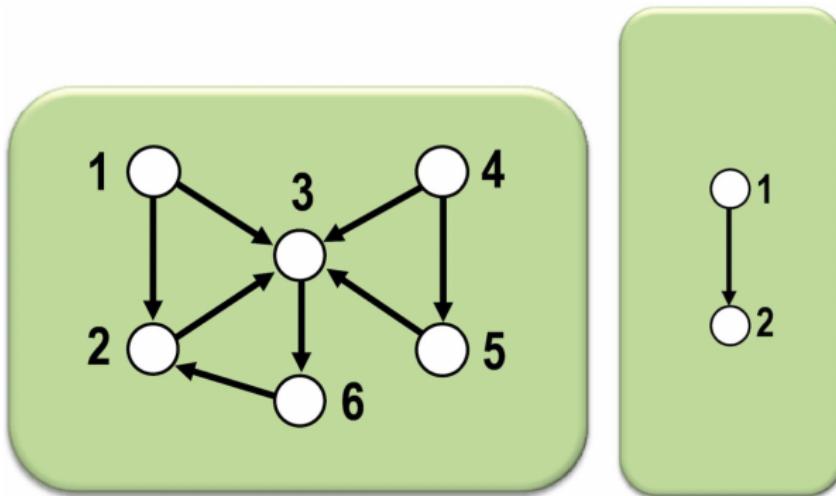


## DFS - Exemplo



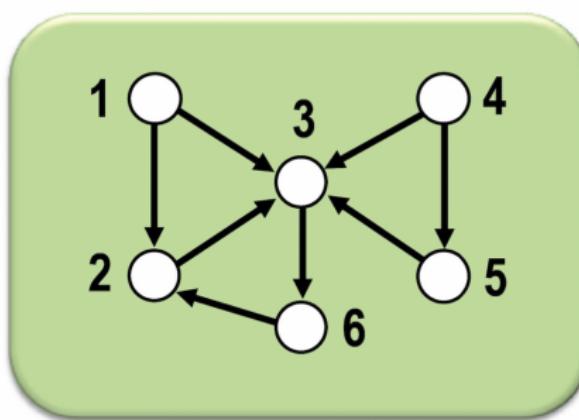
Grafo original.

## DFS - Exemplo

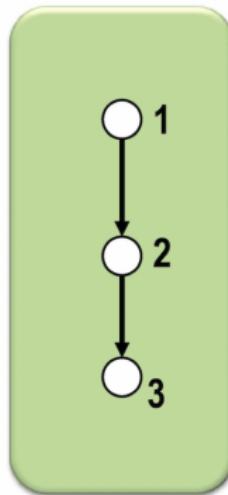


(1) Arco (1,2)

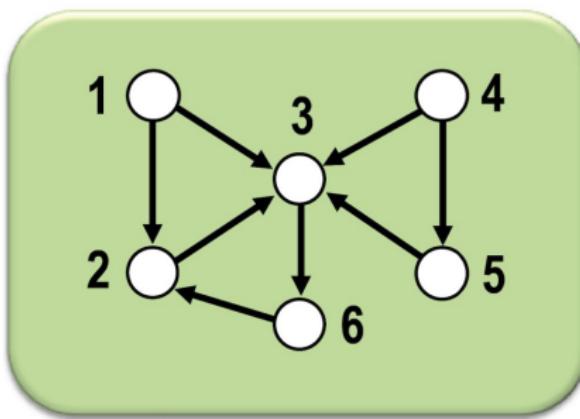
## DFS - Exemplo



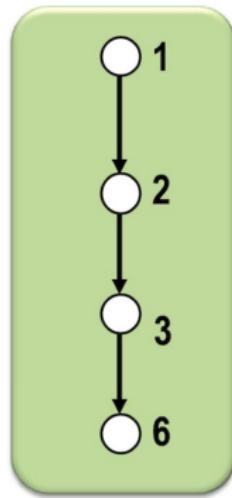
(2) Arco (2,3)



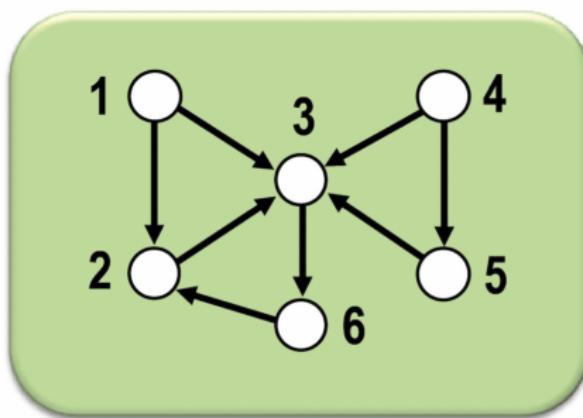
## DFS - Exemplo



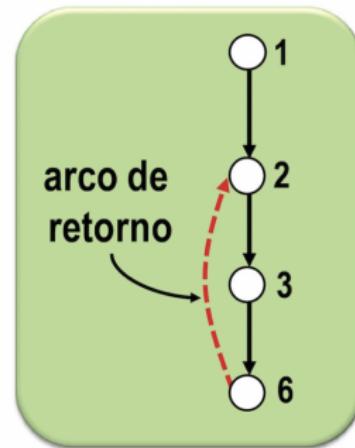
(3) Arco (3,6)



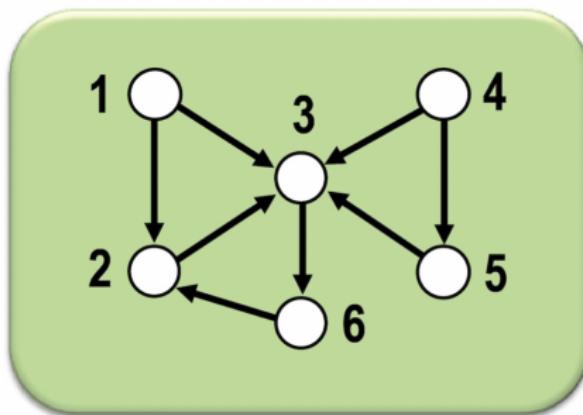
## DFS - Exemplo



(4) Arco (6,2)



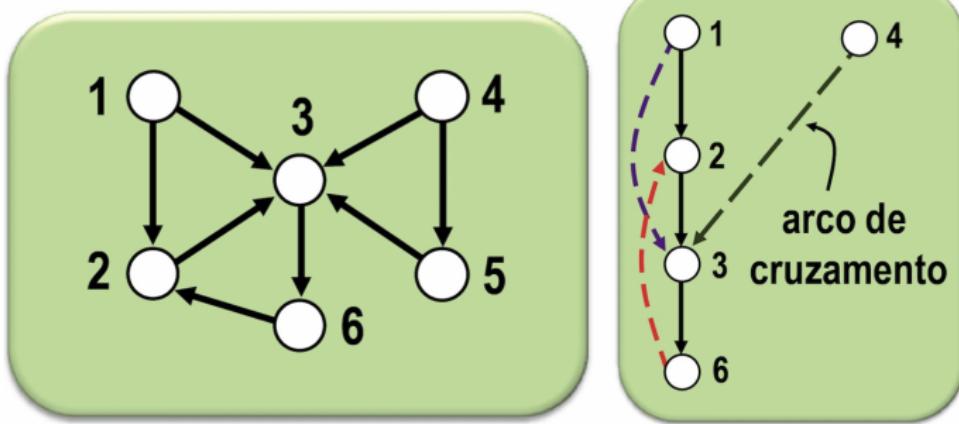
## DFS - Exemplo



(5) Arco (1,3)

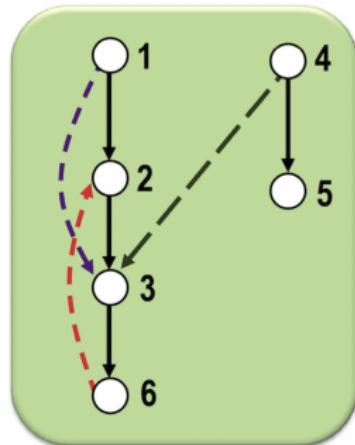
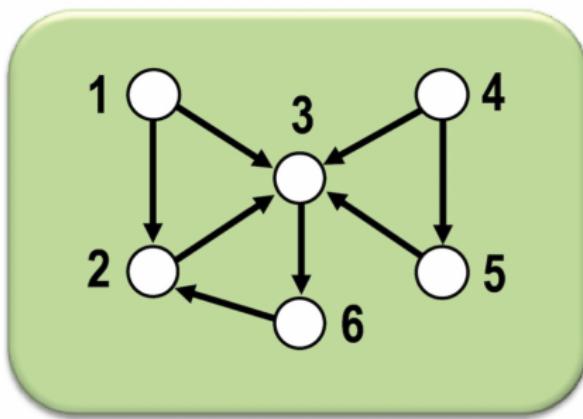


## DFS - Exemplo



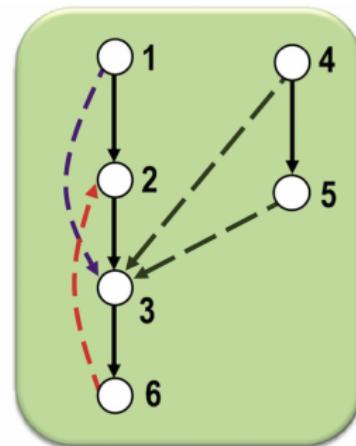
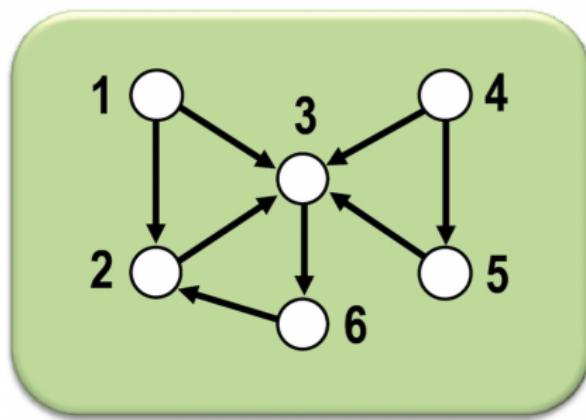
(6) Arco (4,3)

## DFS - Exemplo



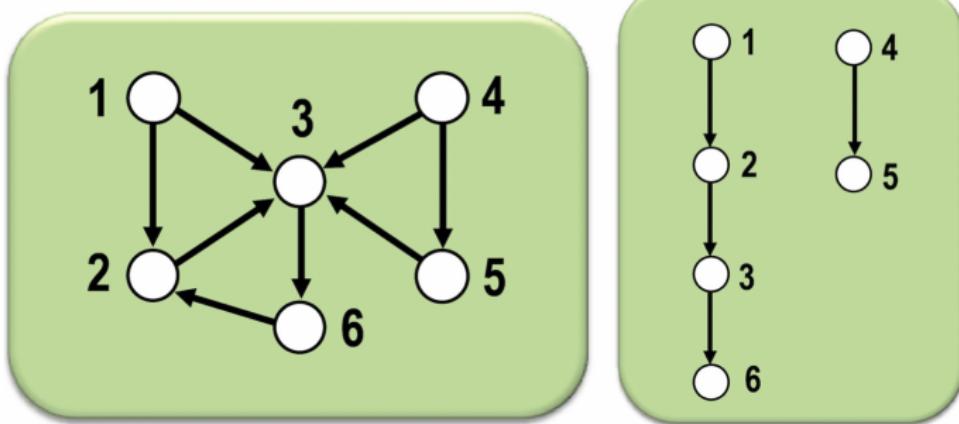
(7) Arco (4,5)

## DFS - Exemplo



(8) Arco (5,3)

## DFS - Exemplo



Grafo original e respectiva floresta de profundidade.

A large, colorful word cloud centered around the words "thank you". The word "thank" is in red, "you" is in green, and "you" is in blue. Numerous other words in different languages are scattered around, such as "danke" (German), "merci" (French), "gracias" (Spanish), "спасибо" (Russian), " obrigado" (Portuguese), "감사합니다" (Korean), and many more. The background is white, and the text is in various colors including red, green, blue, yellow, purple, and orange.

