

# Grafos

*Travessia por profundidade*

**Prof. Edson Alves**

**Faculdade UnB Gama**

## **Definição de travessia**

## Definição de travessia

Uma travessia consiste em visitar todos os nós atingíveis a partir de um vértice de partida  $s$ , em alguma ordem. Cada vértice deve ser visitado exatamente uma vez.

## **Características da travessias**

## Características da travessias

- ★ **Duas travessias são distintas se as ordens de visitação são diferentes**

## Características da travessias

- ★ Duas travessias são distintas se as ordens de visitação são diferentes
- ★ Um grafo conectado com  $N$  vértices tem  $N!$  travessias distintas

## Características da travessias

- ★ Duas travessias são distintas se as ordens de visitação são diferentes
- ★ Um grafo conectado com  $N$  vértices tem  $N!$  travessias distintas
- ★ Travessias notáveis: por profundidade e por extensão

## Travessia por profundidade (*Depth-first search*)



## Travessia por profundidade (*Depth-first search*)

Seja  $s$  o vértice de partida e  $u$  o vértice observado no momento. As regras abaixo definem a DFS:

## Travessia por profundidade (*Depth-first search*)

Seja  $s$  o vértice de partida e  $u$  o vértice observado no momento. As regras abaixo definem a DFS:

1. Faça  $u = s$

## Travessia por profundidade (*Depth-first search*)

Seja  $s$  o vértice de partida e  $u$  o vértice observado no momento. As regras abaixo definem a DFS:

1. **Faça**  $u = s$

2. **Visite**  $u$

## Travessia por profundidade (*Depth-first search*)

Seja  $s$  o vértice de partida e  $u$  o vértice observado no momento. As regras abaixo definem a DFS:

1. Faça  $u = s$

2. Visite  $u$

3.1 Se  $u$  teve ao menos um vizinho  $v$  ainda não visitado, faça  $u = v$

## Travessia por profundidade (*Depth-first search*)

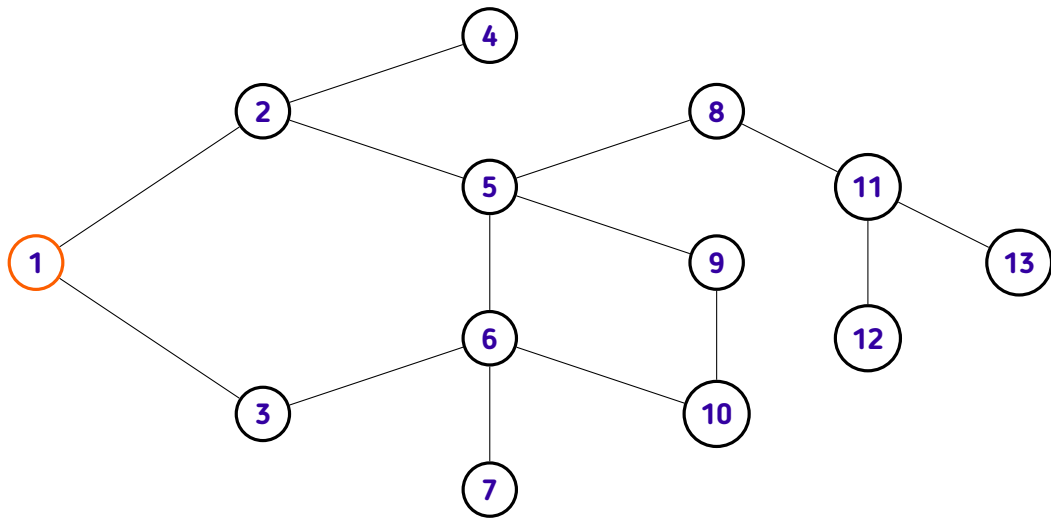
Seja  $s$  o vértice de partida e  $u$  o vértice observado no momento. As regras abaixo definem a DFS:

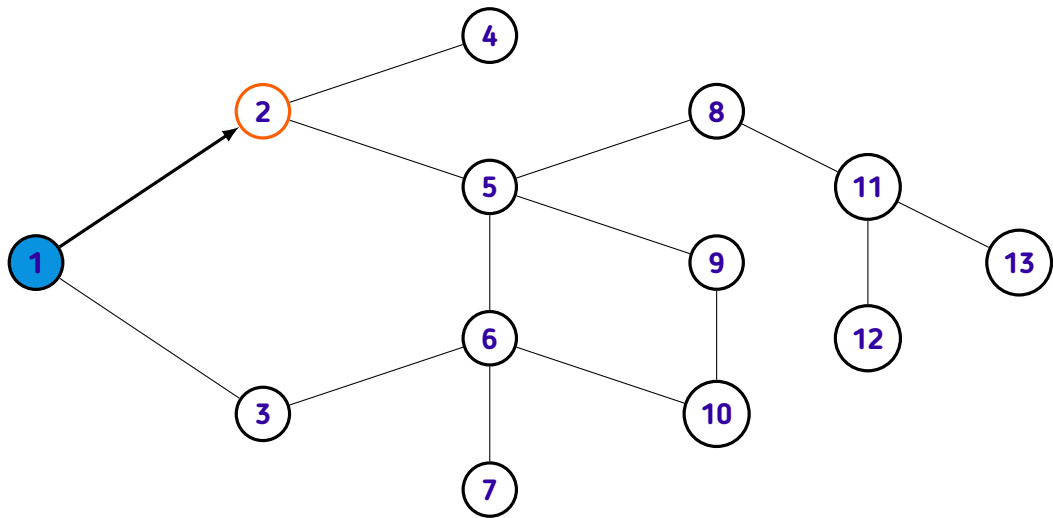
1. Faça  $u = s$

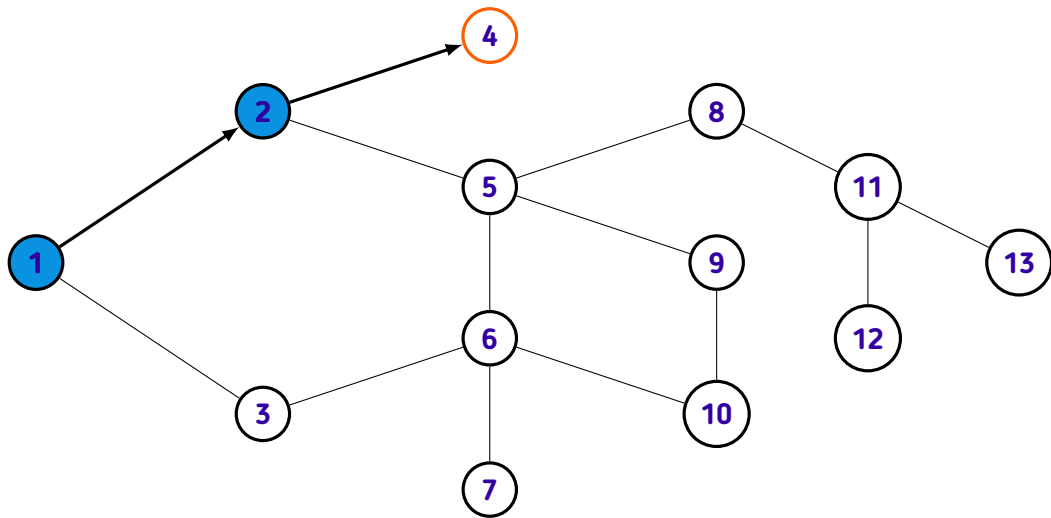
2. Visite  $u$

3.1 Se  $u$  teve ao menos um vizinho  $v$  ainda não visitado, faça  $u = v$

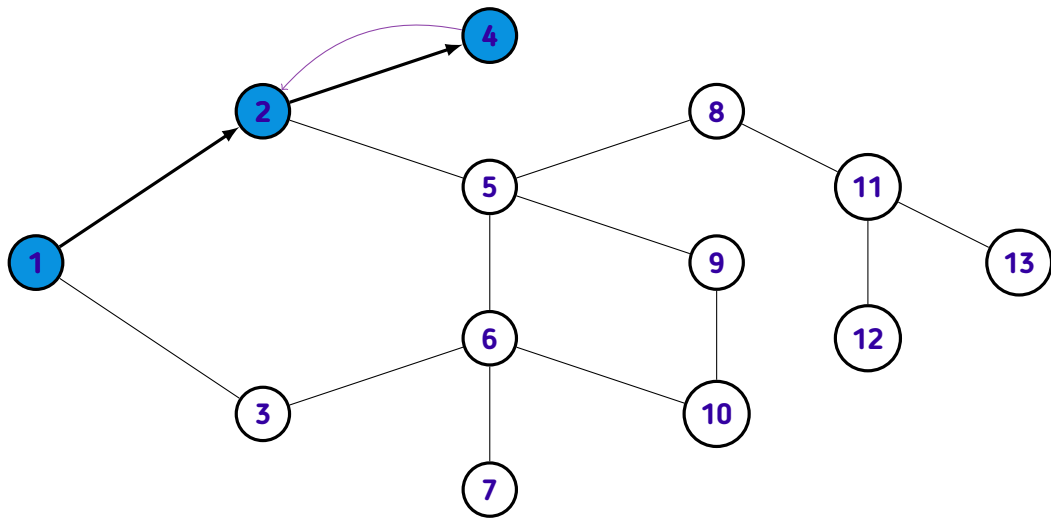
3.2 Caso contrário, volte para o vértice que descobriu  $u$

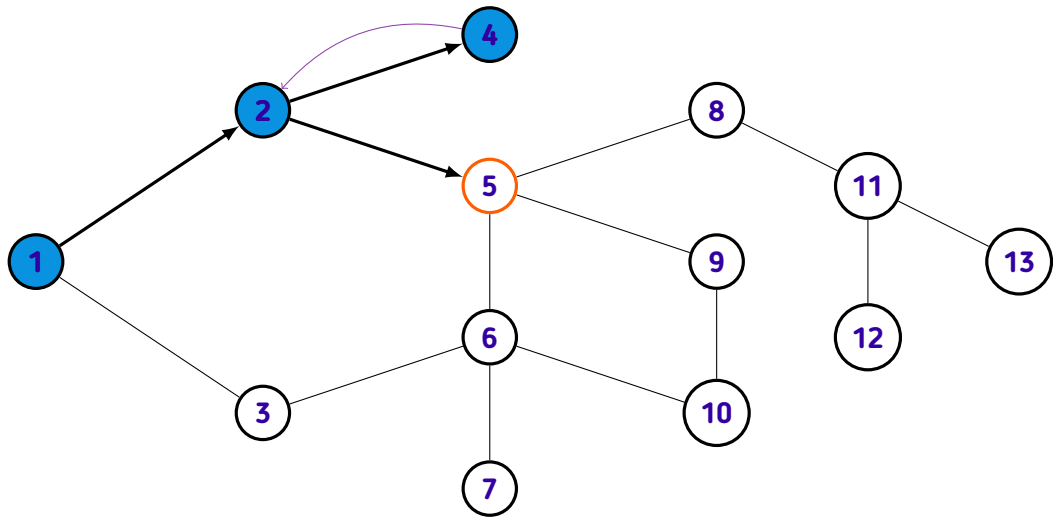


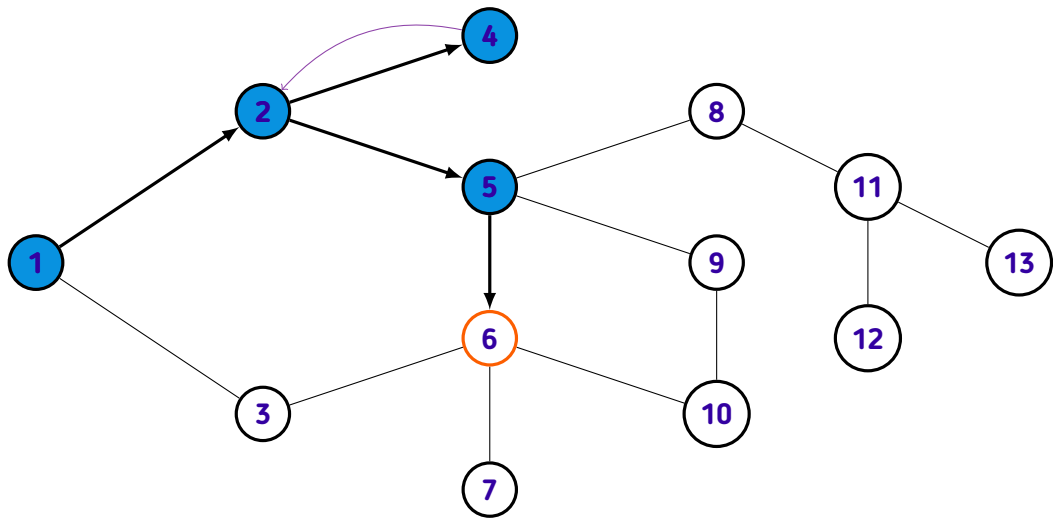


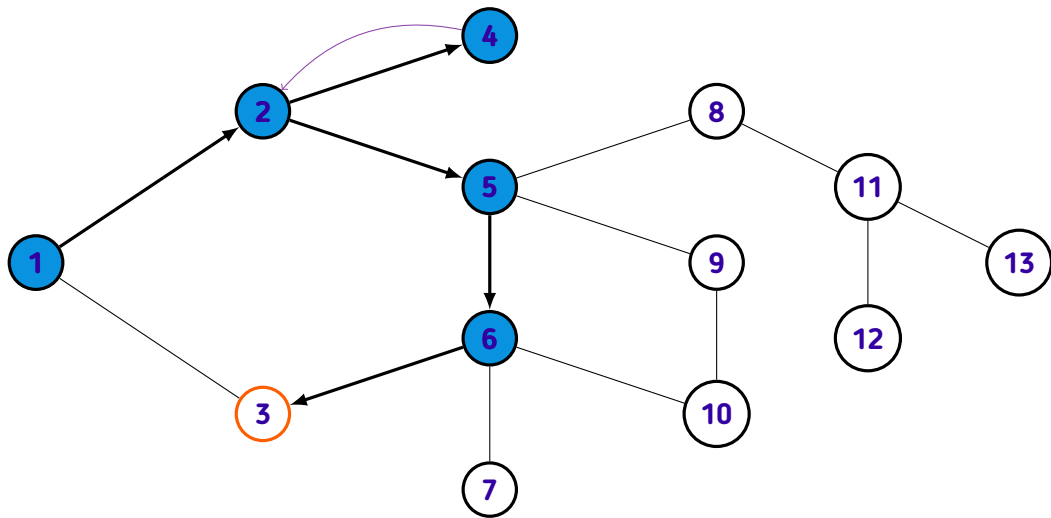


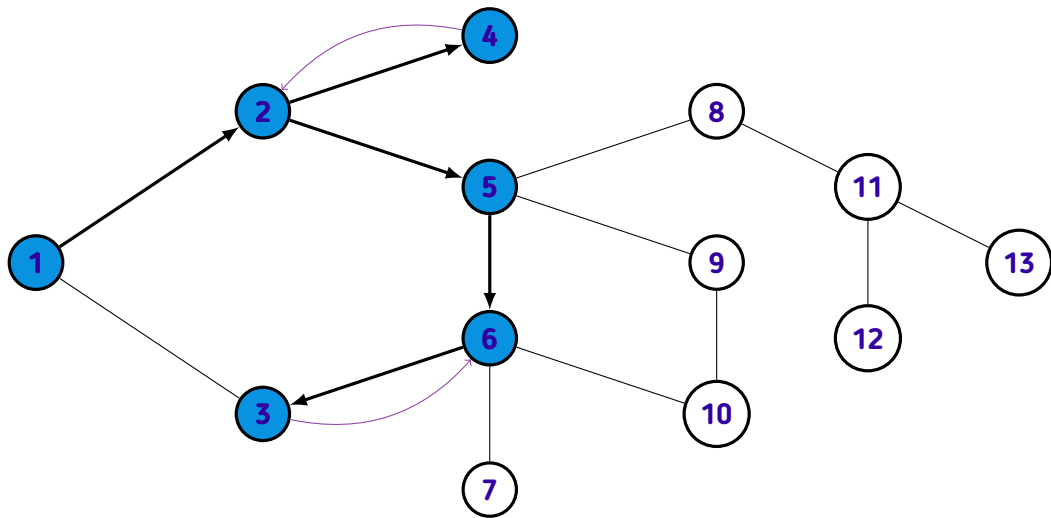


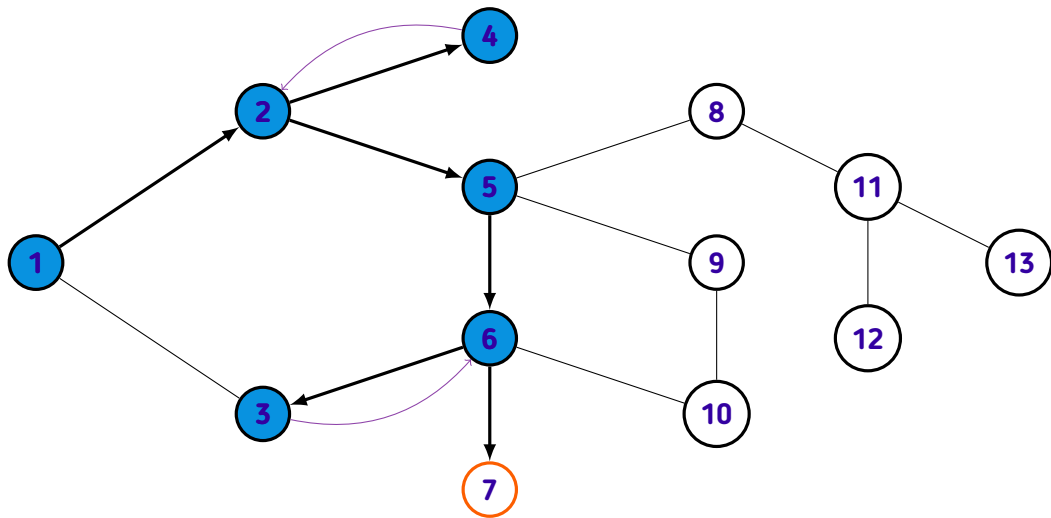


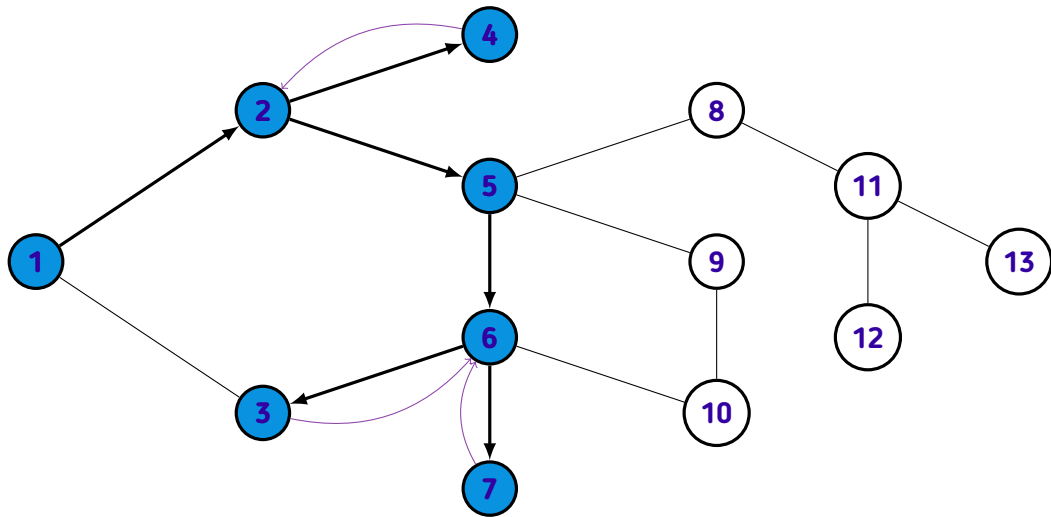


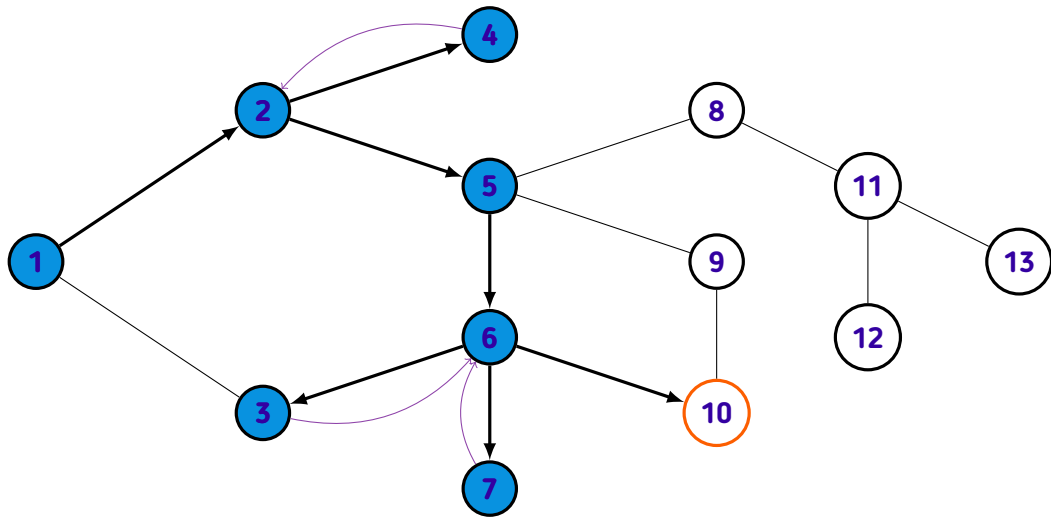




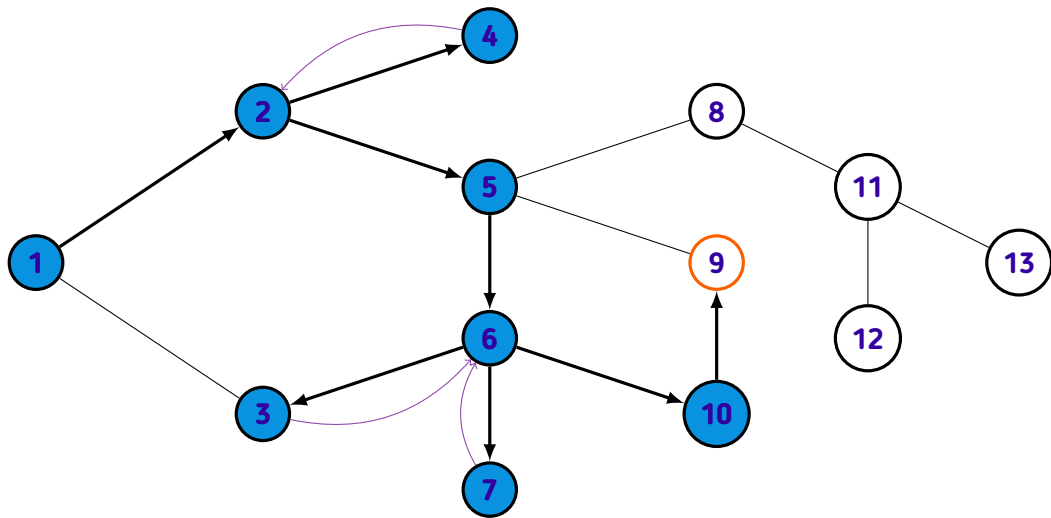


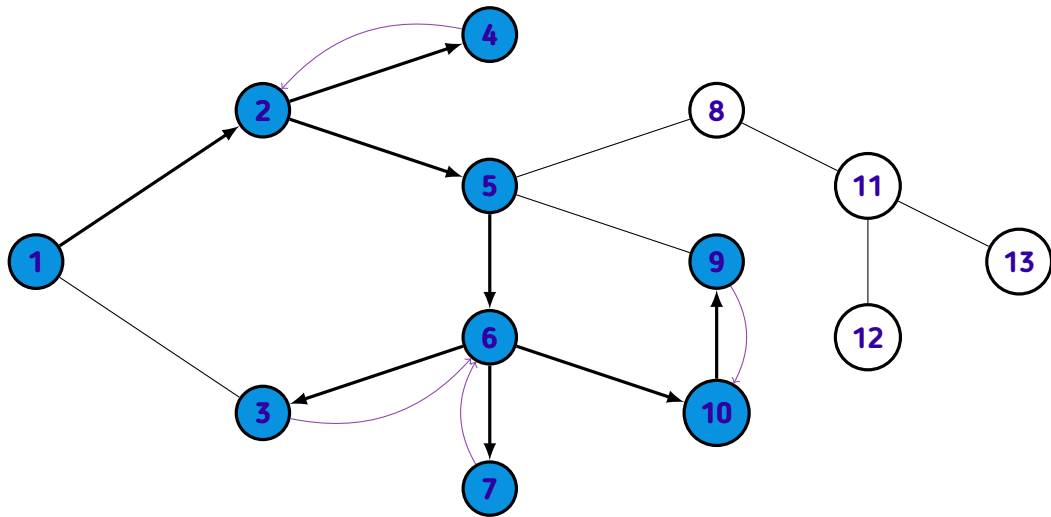


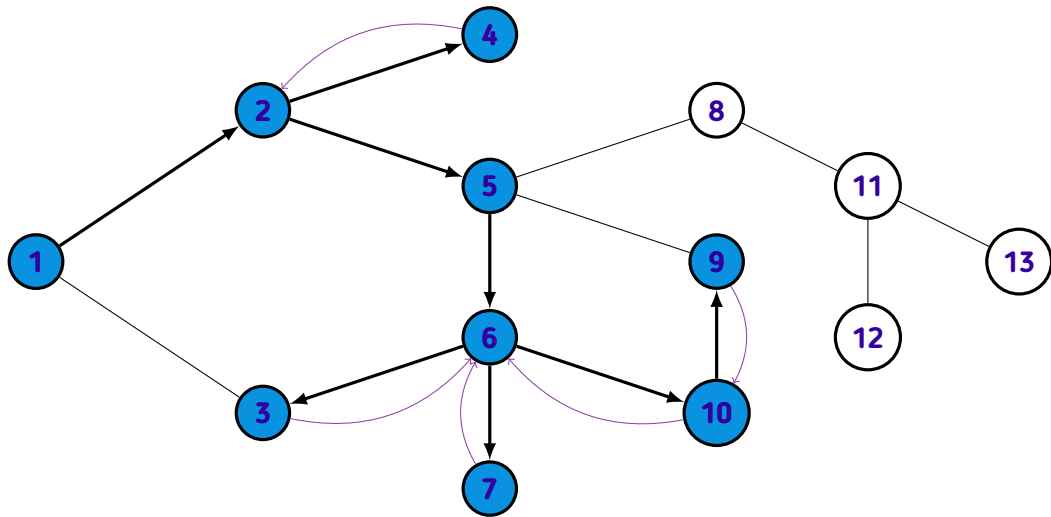


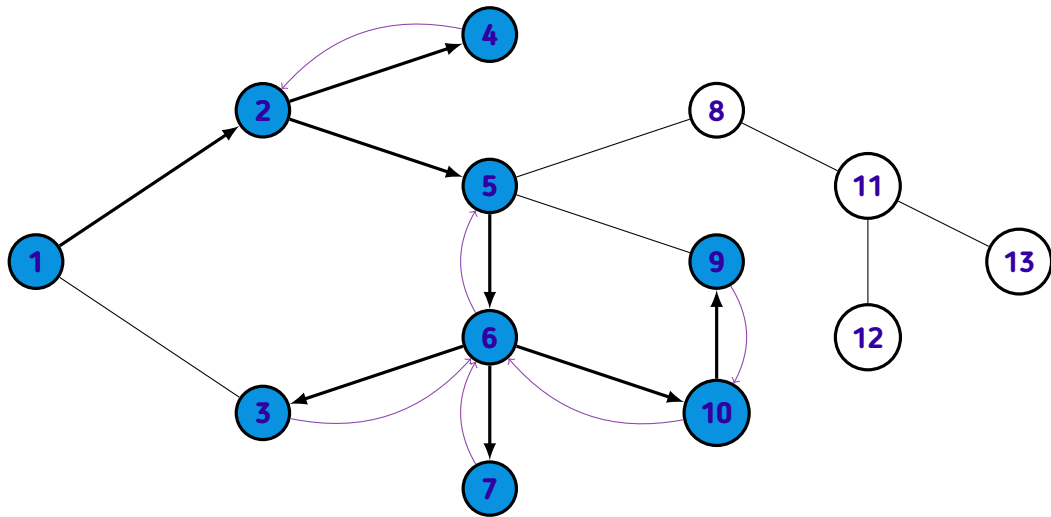


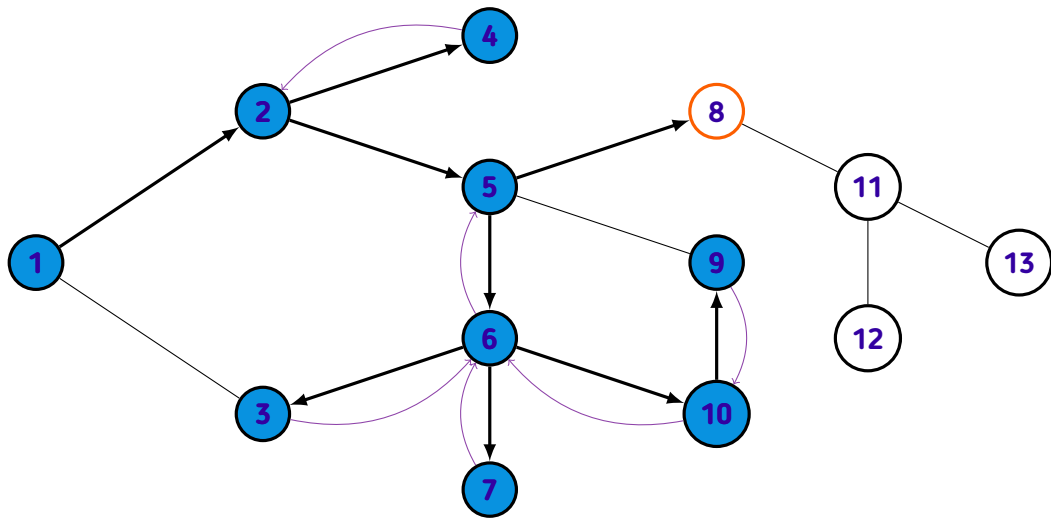


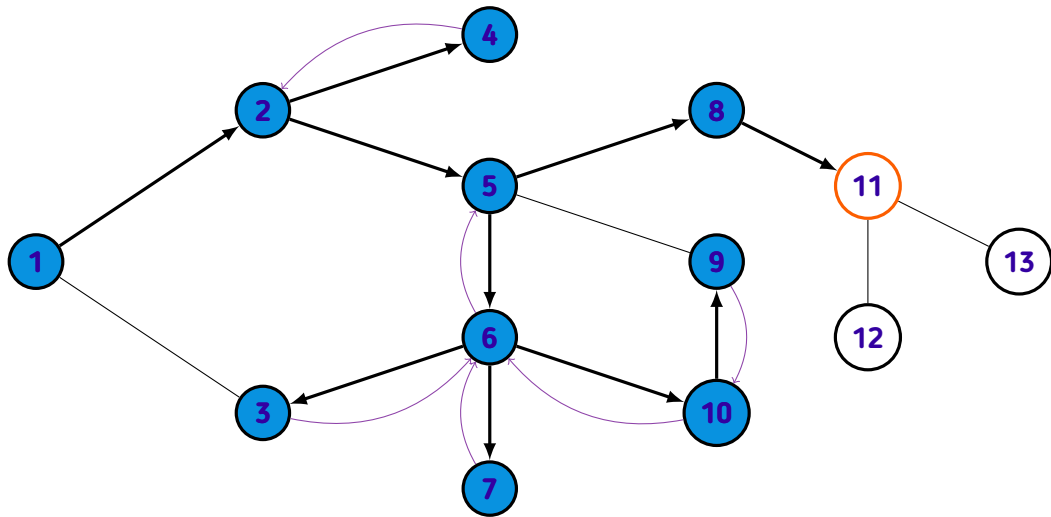


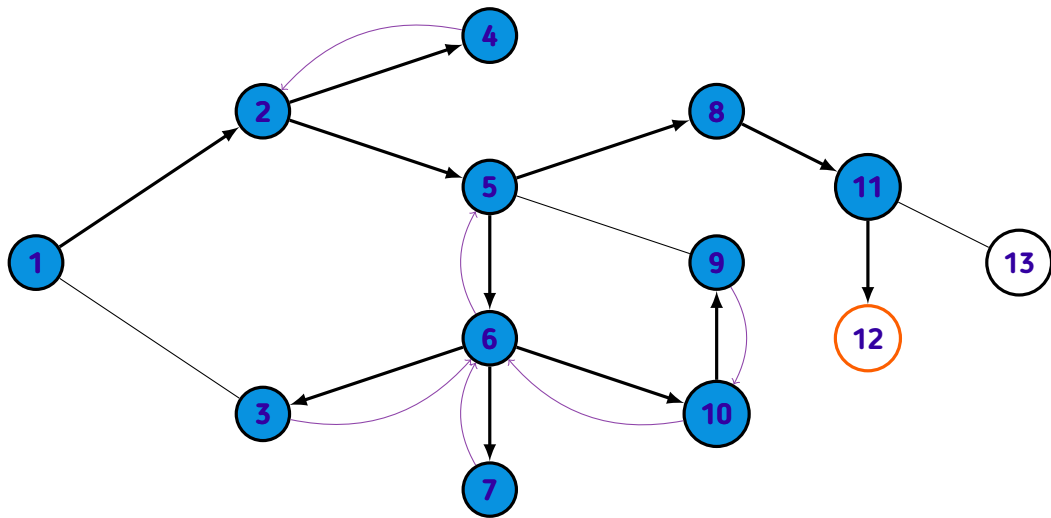


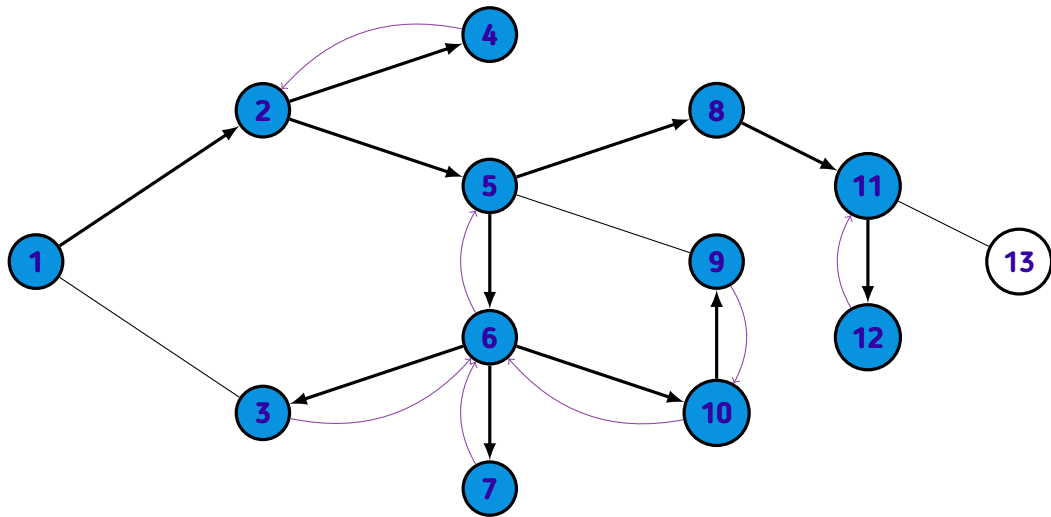




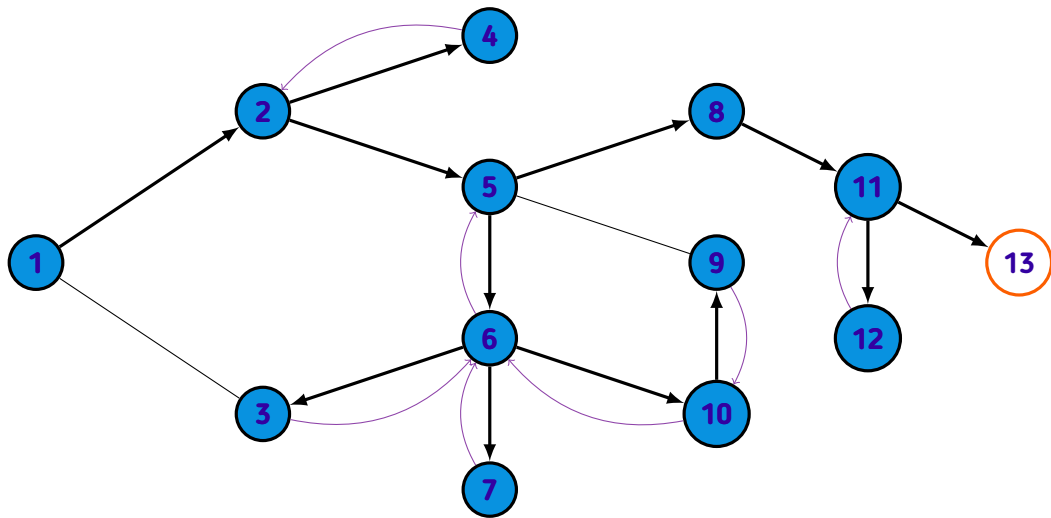


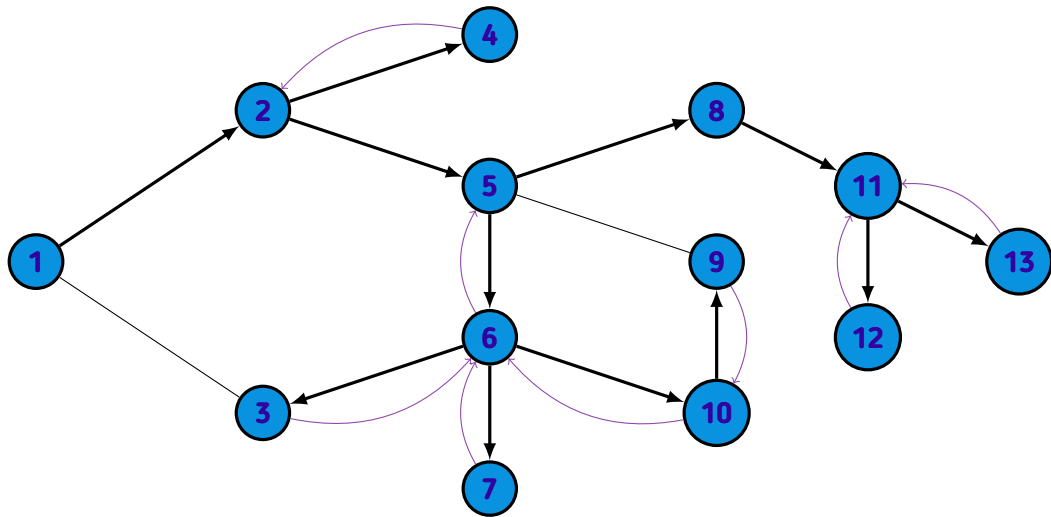


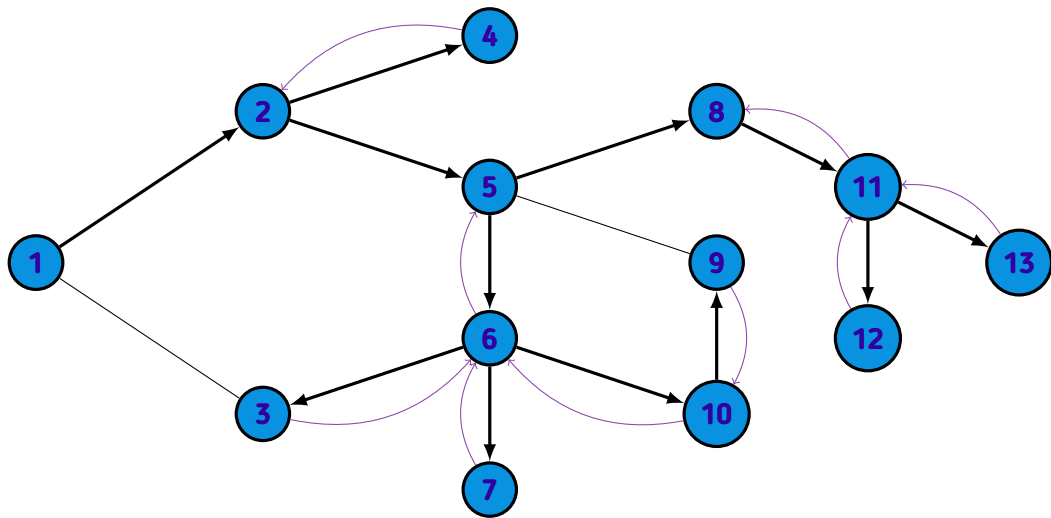


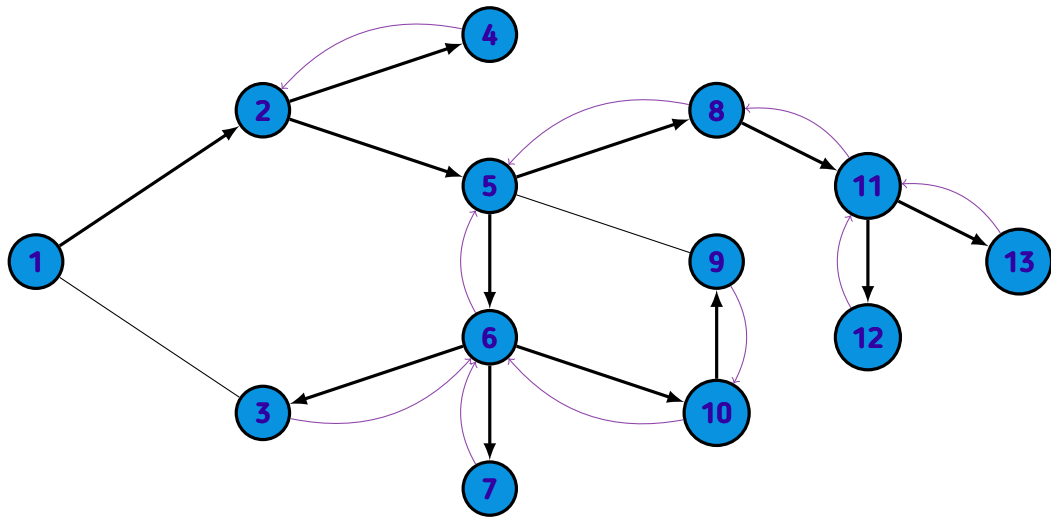


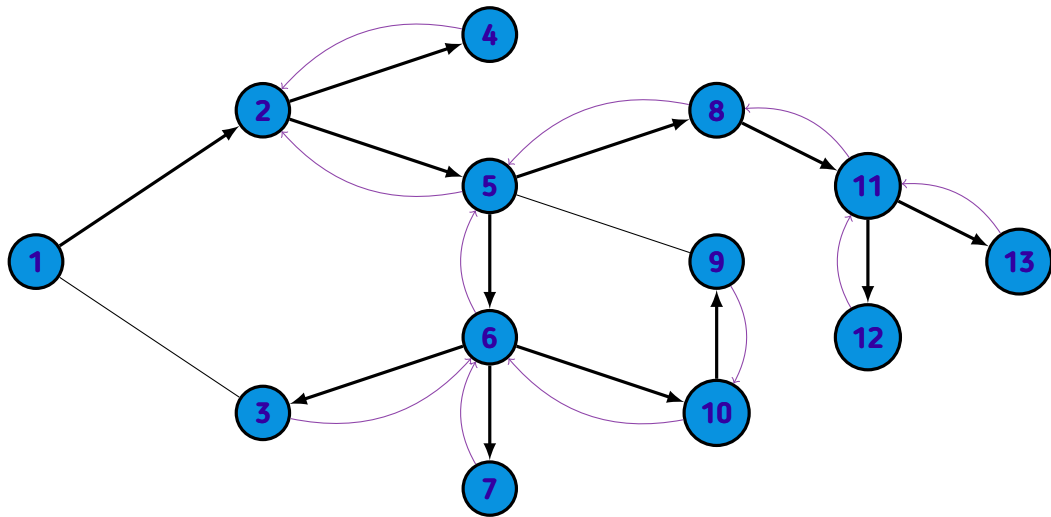


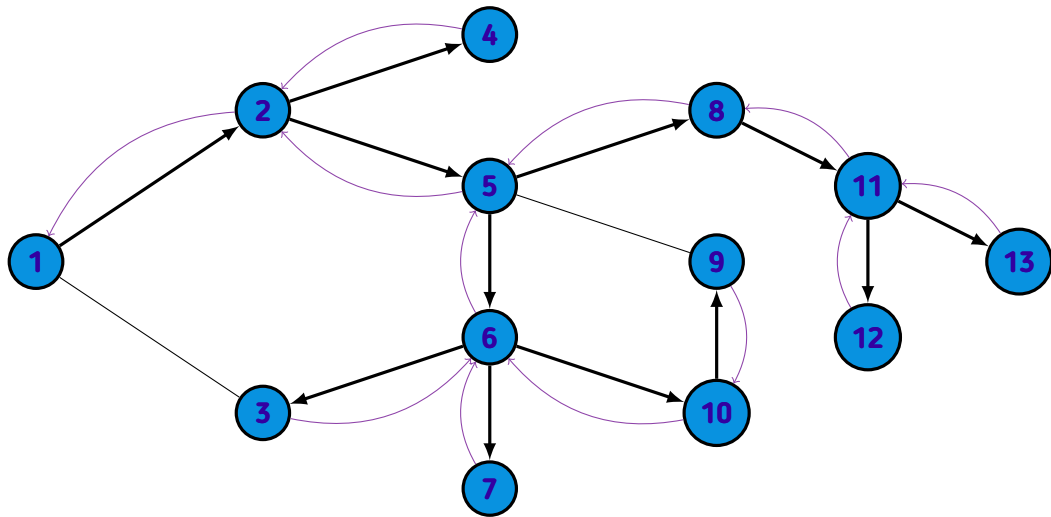












## **Características da DFS**

## Características da DFS

- ★ Cada nó é visitado uma única vez



## Características da DFS

- ★ Cada nó é visitado uma única vez
- ★ Complexidade:  $O(N + M)$  em listas de adjacências

## Características da DFS

- ★ Cada nó é visitado uma única vez
- ★ Complexidade:  $O(N + M)$  em listas de adjacências
- ★ Em matrizes de adjacência a complexidade é  $O(N^2)$