

# ESTRUTURA DE DADOS

Prof.<sup>a</sup> Priscilla Abreu

[priscilla.braz@rj.senac.br](mailto:priscilla.braz@rj.senac.br)



# Estrutura de dados



## Roteiro de Aula

- Objetivo da aula
- Árvores

# Estrutura de dados



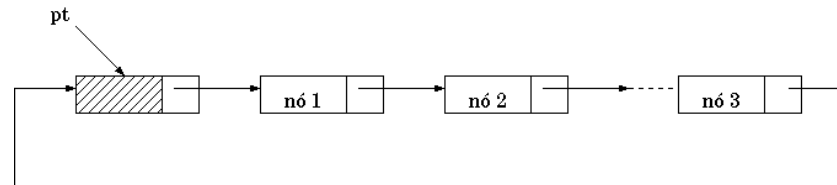
## Objetivo da aula

Identificar situações em que são necessárias o uso de estruturas dinâmicas e encadeadas e compreender a manipulação dessas estruturas.

# REVISANDO...

# Estrutura de dados

## LISTA LINEAR

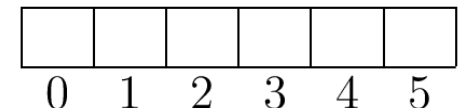


Estrutura que permite representar um conjunto de dados de forma a preservar a relação de ordem existente entre eles.

Uma lista é um exemplo de uma estrutura de dados linear, pois cada elemento tem:

- um predecessor único, exceto o primeiro elemento da lista;
- um sucessor único, exceto o último elemento.

As pilhas e filas são outros exemplos.



## LISTA LINEAR

### Listas lineares

#### Listas lineares gerais

SEM restrição de inserção e remoção de elementos

#### Listas particulares

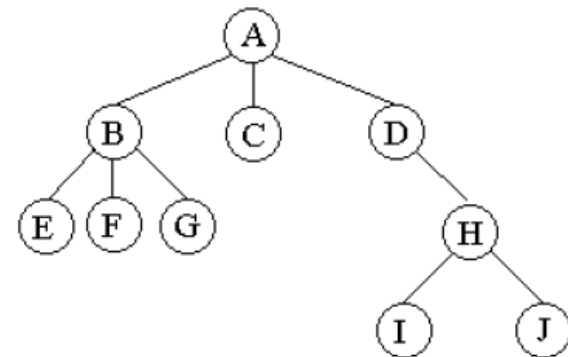
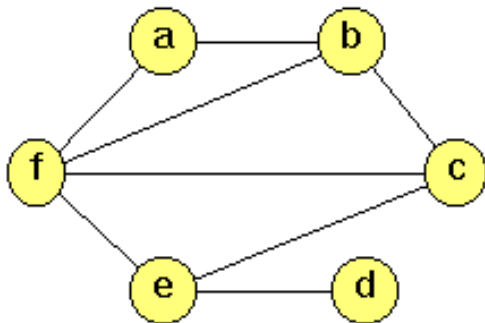
COM restrição de inserção e remoção de elementos

# Estrutura de dados

## LISTAS NÃO LINEARES

Em uma estrutura de dados não linear, os elementos, designados por nós, podem ter mais de um predecessor ou mais de um sucessor.

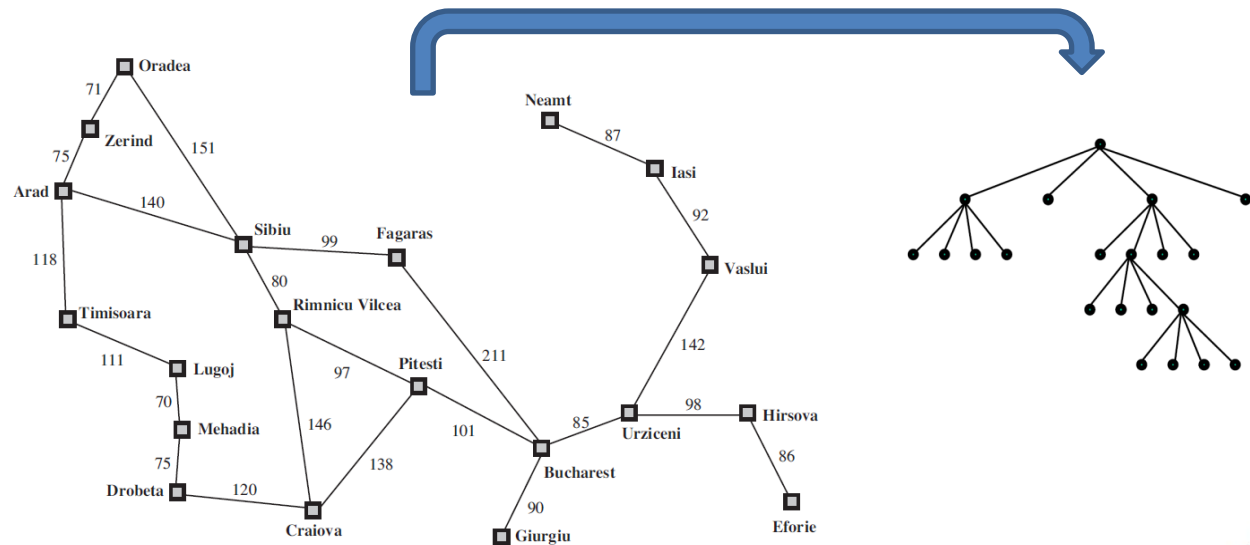
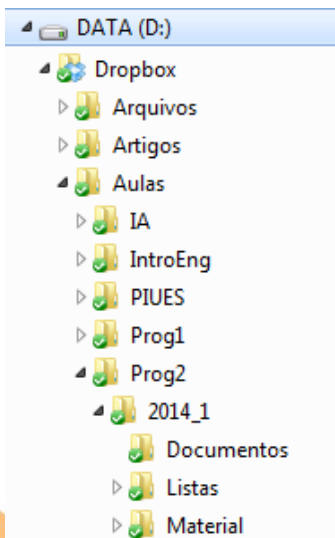
### Grafos e árvores



# Estrutura de dados

## LISTAS NÃO LINEARES – ÁRVORES

Um estrutura de dados do tipo árvore permite que dados sejam organizados de forma hierárquica.

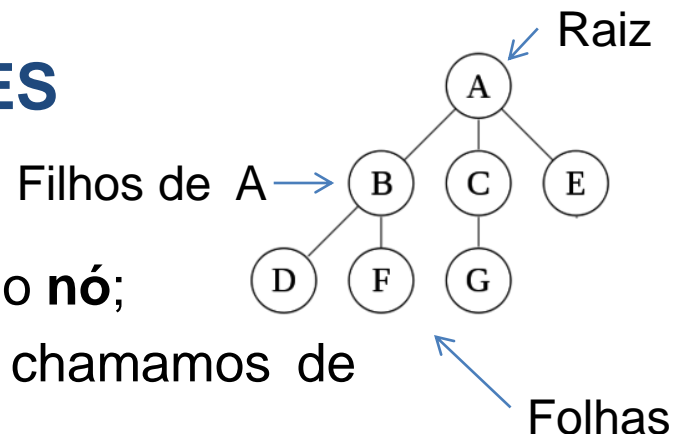




# Estrutura de dados



## LISTAS NÃO LINEARES – ÁRVORES

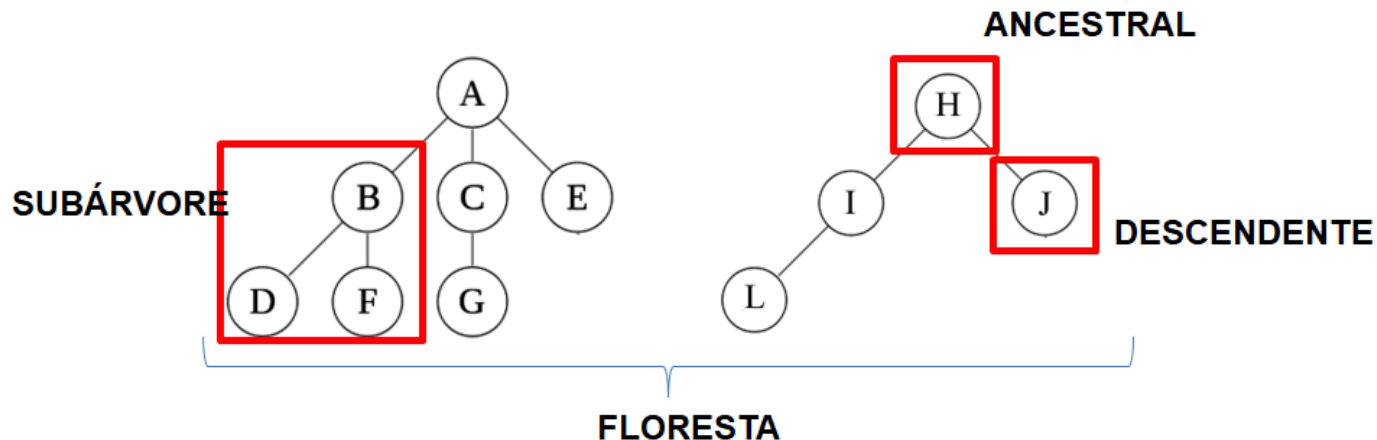


- Cada elemento de uma árvore é denominado **nó**;
- Toda árvore tem um elemento inicial que chamamos de **raiz** da árvore;
- Cada elemento da árvore pode ou não possuir nós abaixo dele hierarquicamente, denominados **filhos**.
- Os nós que não possuem filhos são denominados **folha** ou nó externo.
- Grau de um nó: número de filhos que ele possui.
- Grau da árvore: definido pelo nó de maior grau da árvore.



## ÁRVORES – CONCEITOS

- Subárvore: conjunto de nós formado a partir de um determinado nó.
- Ancestral: nó que antecede um determinado nó.
- Descendente: nó que sucede um determinado nó.
- Floresta: o conjunto de árvores disjuntas.

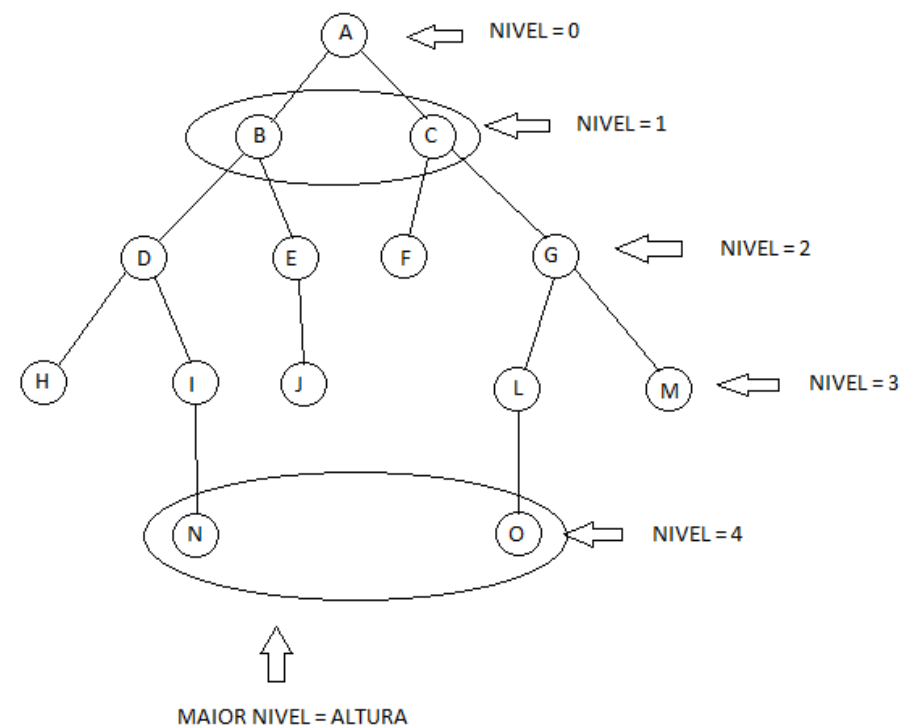
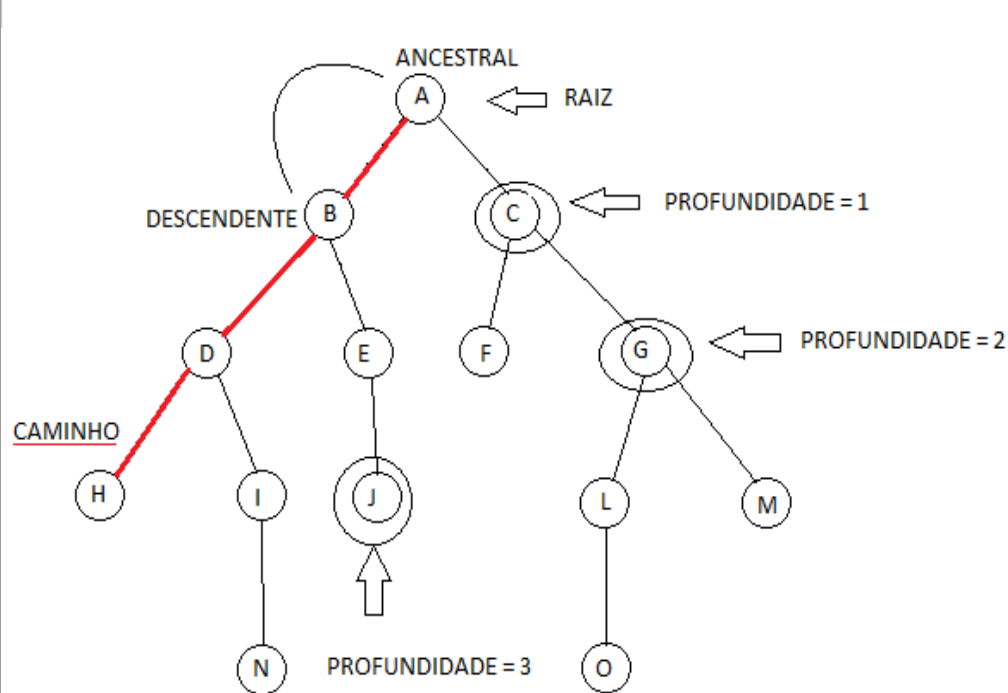


## ÁRVORES – CONCEITOS

- Caminho: sequência de vértices tal que de cada um dos vértices existe uma aresta para o vértice seguinte.
- Profundidade: distancia de um nó até a raiz.
- Nível: conjunto de nós com a mesma profundidade. O nó raiz possui nível 0.
- Altura: comprimento do caminho mais longo da raiz até uma de suas folhas.

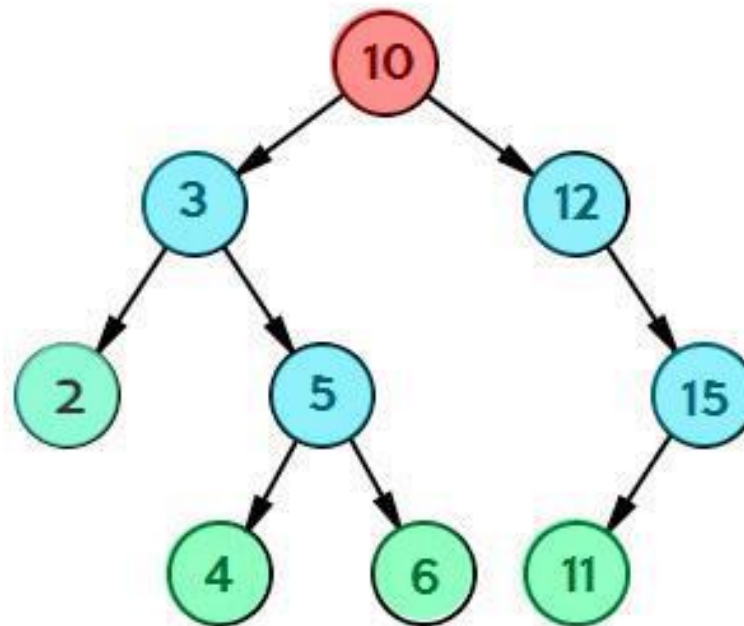
# Estrutura de dados

## ÁRVORES – CONCEITOS



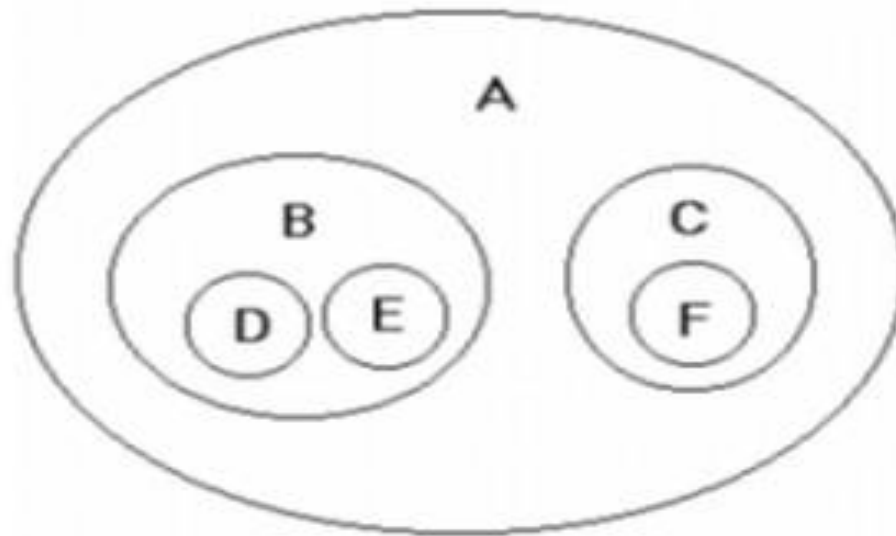
## ÁRVORES – REPRESENTAÇÃO

- Hierárquica



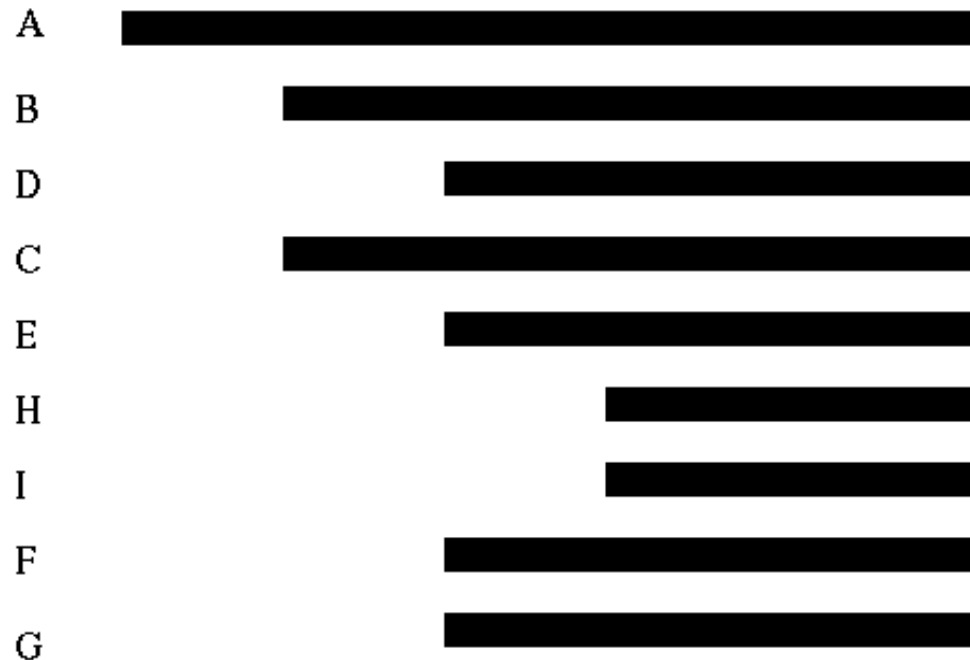
## ÁRVORES – REPRESENTAÇÃO

- Diagrama de Venn



## ÁRVORES – REPRESENTAÇÃO

- Gráfico de Barras



## ÁRVORES – REPRESENTAÇÃO

- Expressão com parênteses

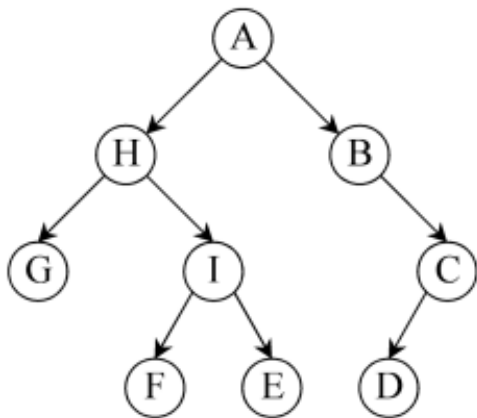
**( A ( B ( D ( ) E ( ) ) ) ( C ( F ( ) ) ) ) )**



# Estrutura de dados

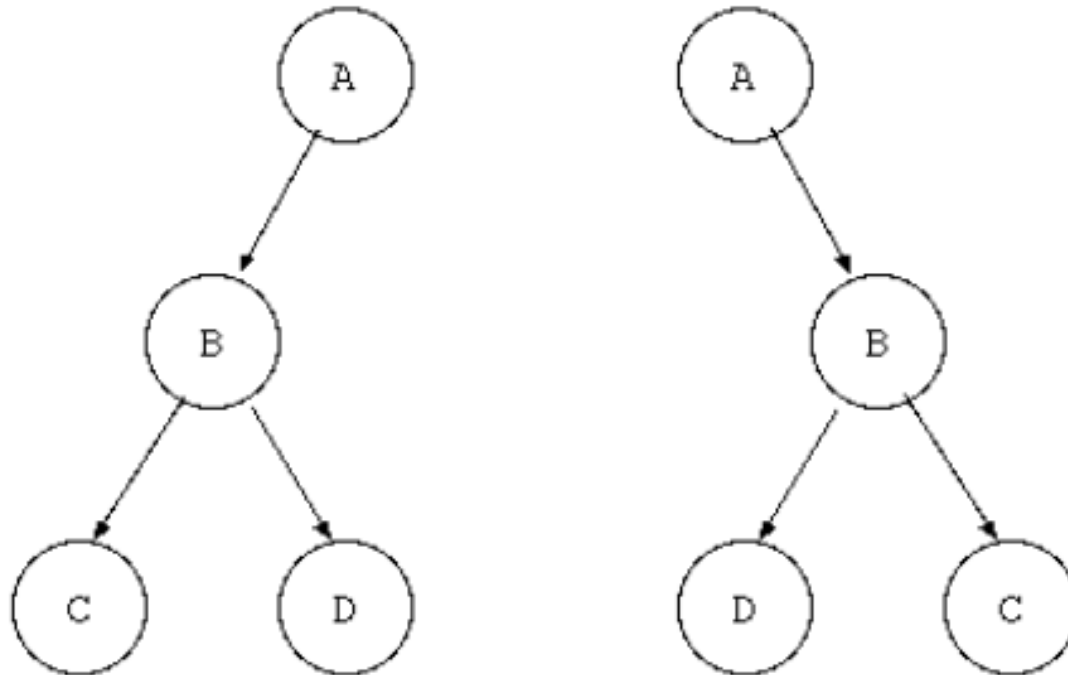
## ÁRVORE BINÁRIA

Estrutura de dados que é constituída por um conjunto finito de nós, em que cada nó pode ter no máximo dois filhos, ou sub-árvores: a sub-árvore da direita (sad) e a sub-árvore da esquerda (sae).



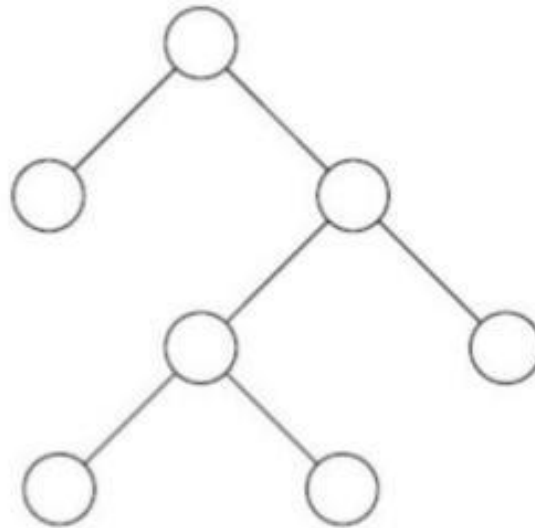
# Estrutura de dados

## ÁRVORE X ÁRVORE BINÁRIA



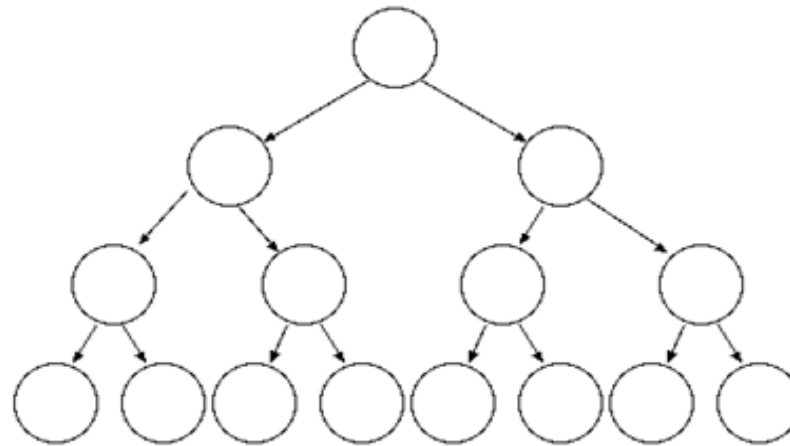
## ÁRVORE ESTRITAMENTE BINÁRIA

Árvore binária em que cada nó tem 0 ou 2 filhos.



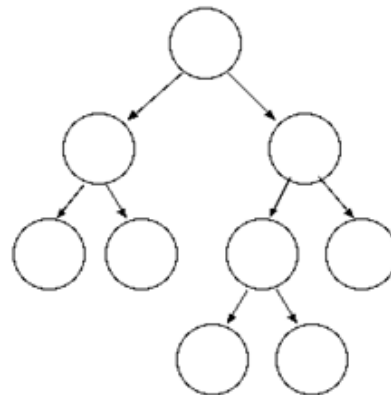
## ÁRVORE BINÁRIA CHEIA

Árvore estritamente binária em que se um nó tem alguma subárvore vazia então ele está no último nível.



## ÁRVORE BINÁRIA COMPLETA

Árvore binária completa: árvore em que se  $n$  é um nó com algumas de suas subárvores vazias, então  $n$  se localiza no penúltimo ou no último nível. Portanto, toda árvore cheia é completa e estritamente binária.



## ÁRVORE BINÁRIA – PERCURSO

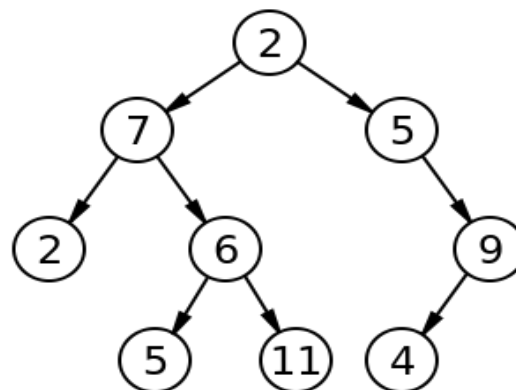
- PRÉ ORDEM
- EM ORDEM
- PÓS ORDEM

## ÁRVORE BINÁRIA – PERCURSO

- PRÉ ORDEM

No percurso em pré-ordem, primeiramente a raiz é visitada; depois, a sub-árvore esquerda; e finalmente, a sub-árvore direita.

No exemplo, o percurso seria feito na seguinte ordem: 2, 7, 2, 6, 5, 11, 5, 9 e 4.

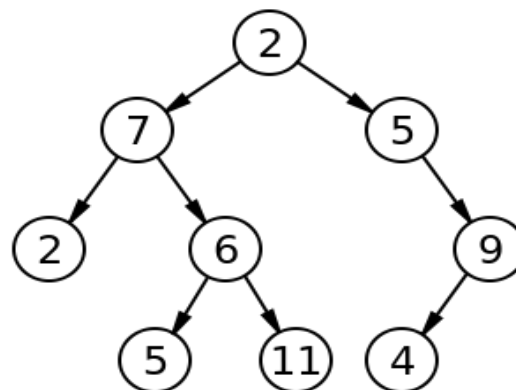


## ÁRVORE BINÁRIA – PERCURSO

- EM ORDEM (SIMÉTRICO)

No percurso simétrico (em ordem), primeiro é visitada a sub-árvore esquerda; logo após, a raiz; por final, a sub-árvore direita.

No exemplo, o percurso seria feito na seguinte ordem: 2, 7, 5, 6, 11, 2, 5, 4 e 9.



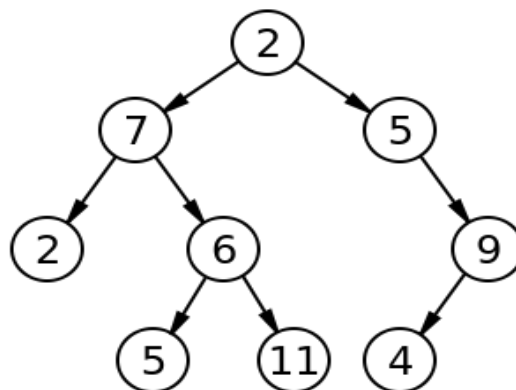


## ÁRVORE BINÁRIA – PERCURSO

- PÓS ORDEM

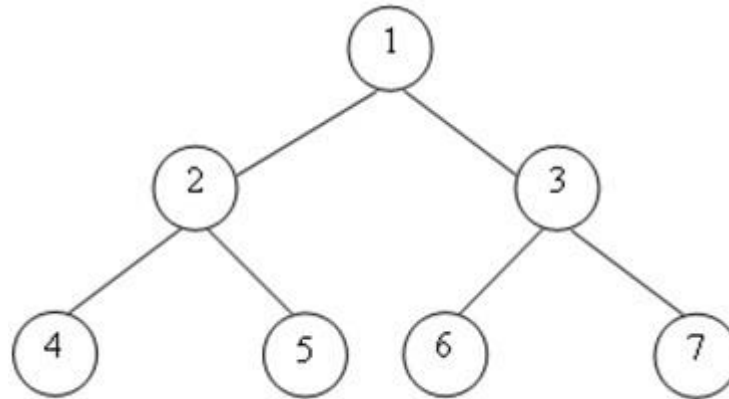
O percurso em pós-ordem inicia-se visitando a sub-árvore esquerda; em seguida, a sub-árvore direita; encerrando, a raiz é visitada.

No exemplo, o percurso seria feito na seguinte ordem: 2, 5, 11, 6, 7, 4, 9, 5 e 2.



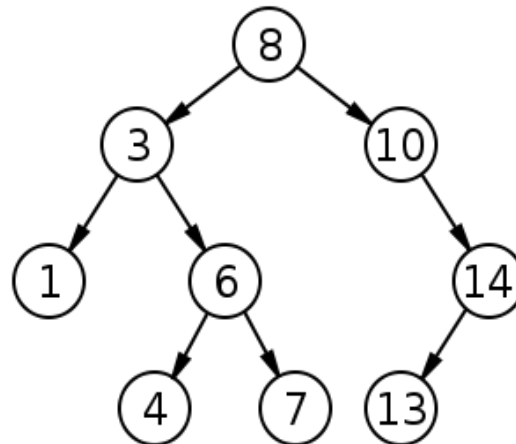
## ÁRVORE BINÁRIA – PERCURSO

Exercício: Qual a ordem do percurso da árvore abaixo se utilizarmos o percurso simétrico?



## ÁRVORE BINÁRIA DE BUSCA

Árvore binária baseada em nós, onde todos os nós da subárvore esquerda possuem um valor numérico inferior ao nó raiz e todos os nós da subárvore direita possuem um valor superior ao nó raiz.



# Estrutura de dados



## ÁRVORE BINÁRIA DE BUSCA

### INSERÇÃO

Considere que iremos inserir os seguintes elementos nessa ordem: 50 60 30 40 70 20

# Estrutura de dados

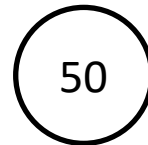


## ÁRVORE BINÁRIA DE BUSCA

### INSERÇÃO

Considere que iremos inserir os seguintes elementos nessa ordem: **50** 60 30 40 70 20

Nó Raiz

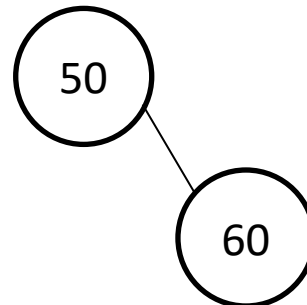


## ÁRVORE BINÁRIA DE BUSCA

### INSERÇÃO

Considere que iremos inserir os seguintes elementos nessa ordem: 50 **60** 30 40 70 20

Nó Raiz



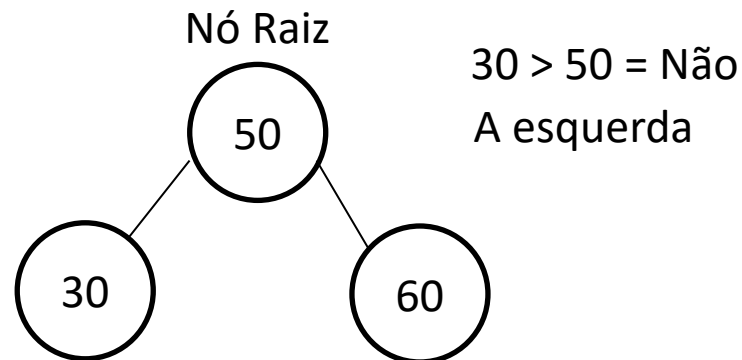
$60 > 50 = \text{SIM}$

A direita

## ÁRVORE BINÁRIA DE BUSCA

### INSERÇÃO

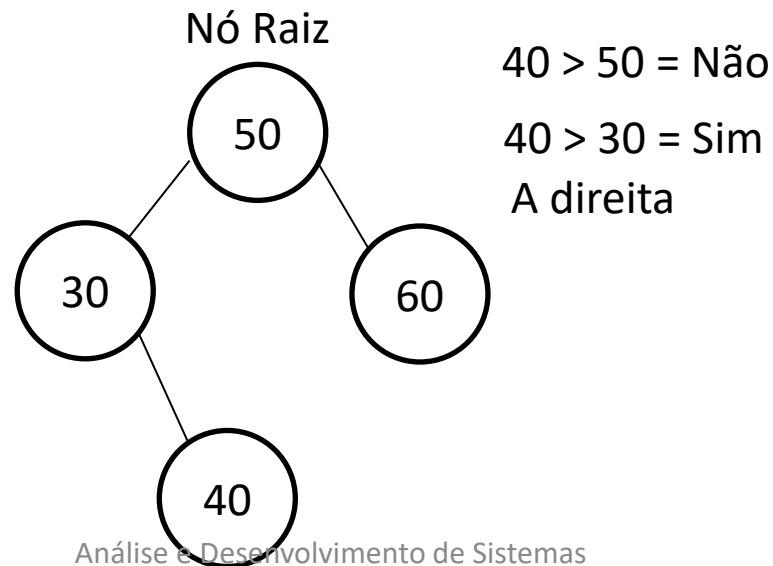
Considere que iremos inserir os seguintes elementos nessa ordem: 50 60 **30** 40 70 20



## ÁRVORE BINÁRIA DE BUSCA

### INSERÇÃO

Considere que iremos inserir os seguintes elementos nessa ordem: 50 60 30 **40** 70 20

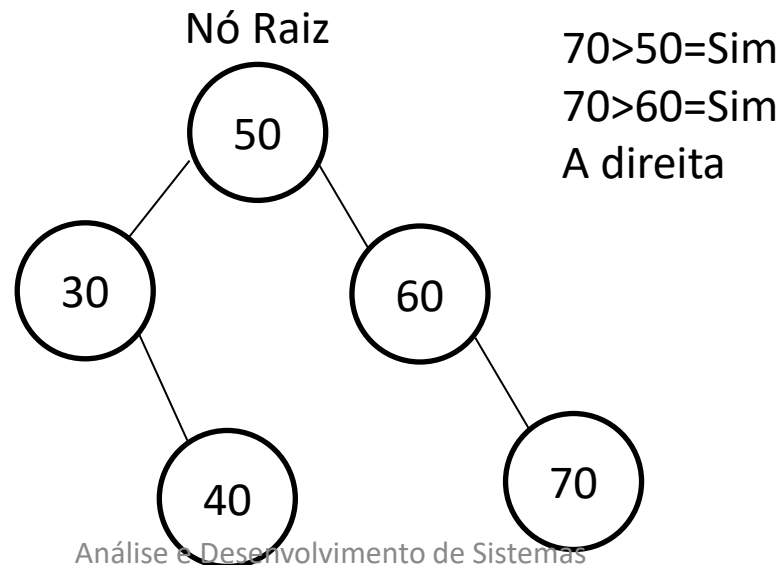




## ÁRVORE BINÁRIA DE BUSCA

### INSERÇÃO

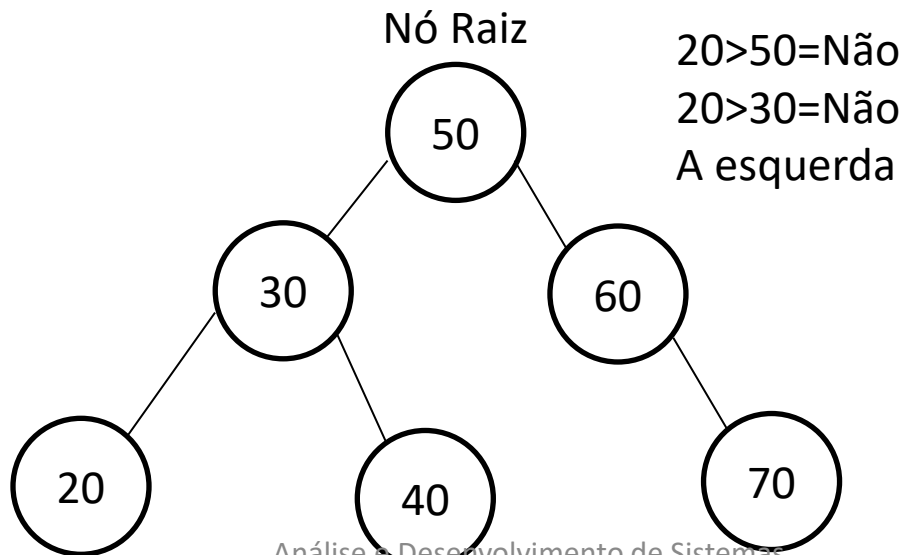
Considere que iremos inserir os seguintes elementos nessa ordem: 50 60 30 40 **70** 20



## ÁRVORE BINÁRIA DE BUSCA

### INSERÇÃO

Considere que iremos inserir os seguintes elementos nessa ordem: 50 60 30 40 70 **20**



# DÚVIDAS???