

# Python Estrutura de Dados

Módulo 07  
Aula 06

# Estrutura de Dados

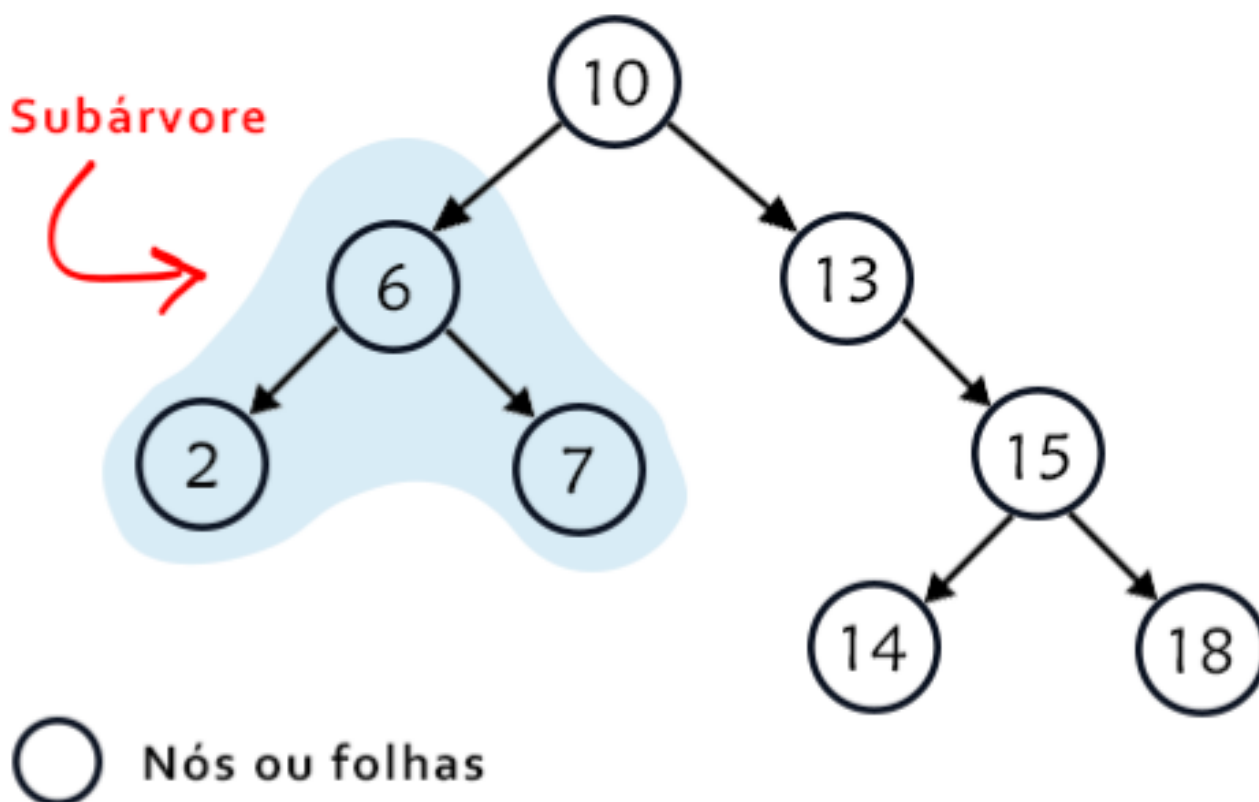
## Árvores Binárias

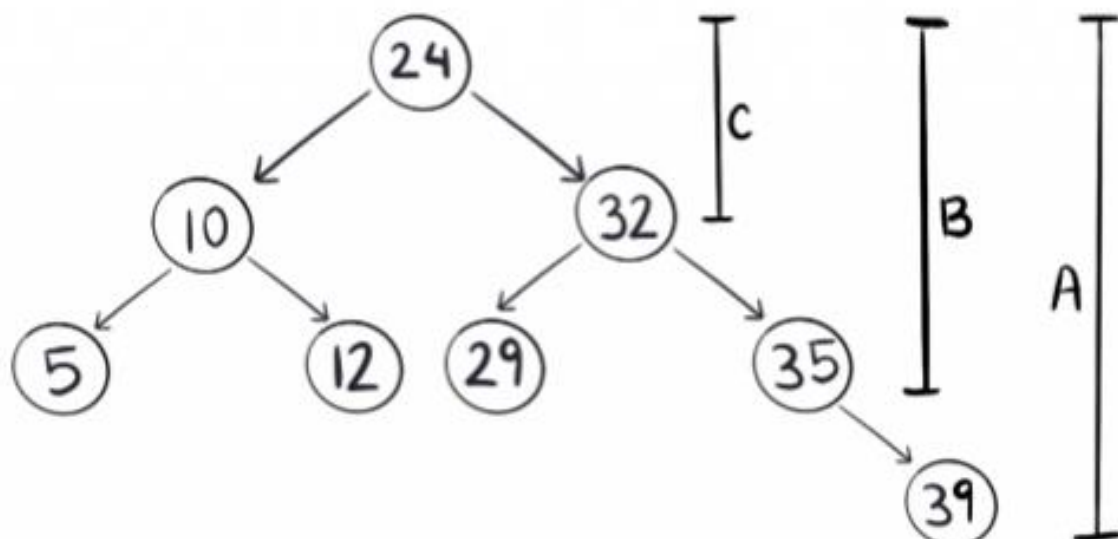
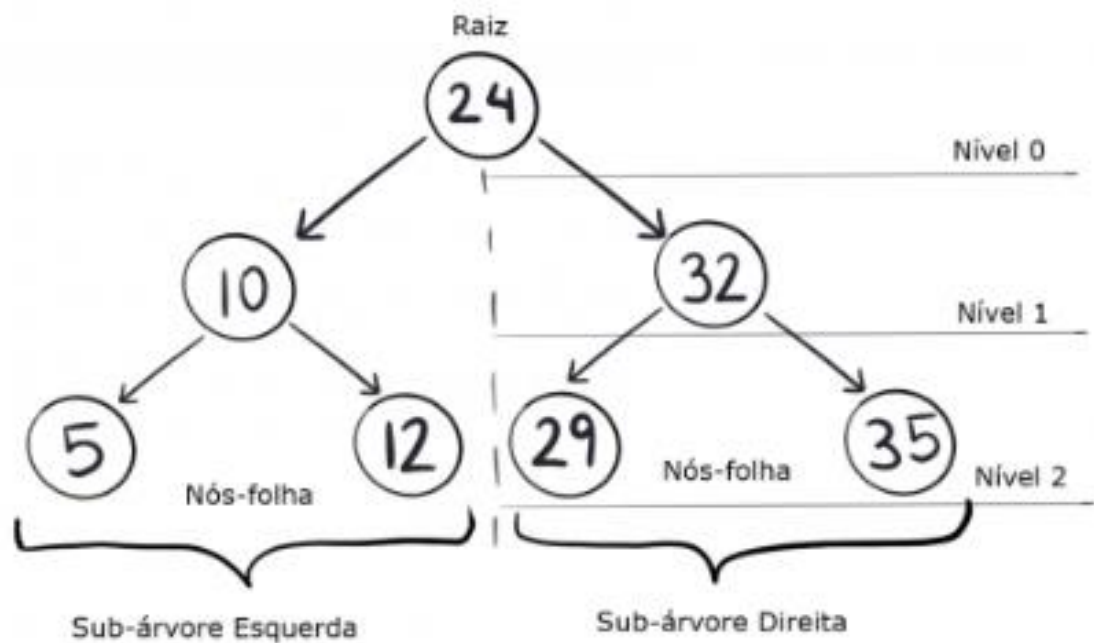
Uma árvore binária é uma estrutura de dados hierárquica não linear, caracterizada por nós onde cada nó pode ter no máximo dois filhos. Estes filhos são referidos como filho esquerdo e filho direito.

Árvores binárias são fundamentais em diversas áreas da computação, incluindo organização de dados, processamento de informações e algoritmos de busca e ordenação. Cada nó em uma árvore binária contém três partes essenciais: um valor ou dado, um ponteiro para o filho esquerdo, e um ponteiro para o filho direito. O nó no topo da árvore é conhecido como a raiz.

- **Subárvores:** Cada filho de um nó em uma árvore binária pode ser a raiz de sua própria subárvore.
- **Folhas:** Nós sem filhos são chamados de folhas ou nós terminais.
- **Altura e Profundidade:** Conceitos importantes para entender a estrutura e navegação em uma árvore binária.

Árvores binárias são a base para estruturas mais complexas, como Árvores Binárias de Busca (BSTs), Heaps e Árvores AVL, cada uma adequada para aplicações específicas em sistemas de computação.





A é a altura do nó raiz (24) até o nó-folha (39), que é 3  
 B é a altura do nó raiz (24) até o nó (35), que é 2  
 C é a altura do nó raiz (24) até o nó-filho (32), que é 1



## ALGORITMO DE CAMINHAMENTO

caminhamento pre-ordem:

- visita nodo corrente
- visita filho da esquerda
- visita filho da direita

caminhamento em ordem:

- visita filho da esquerda
- visita nodo corrente
- visita filho da direita

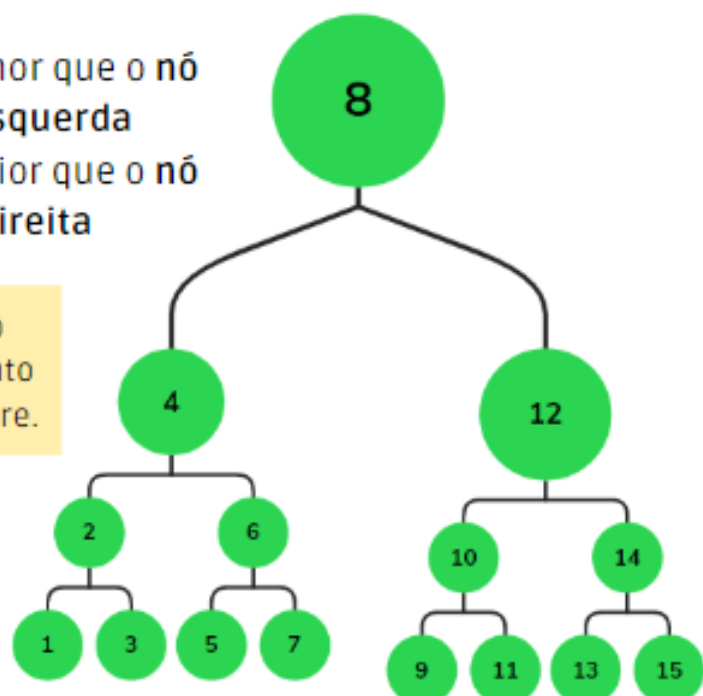
caminhamento pos-ordem:

- visita filho da esquerda
- visita filho da direita
- visita nodo corrente

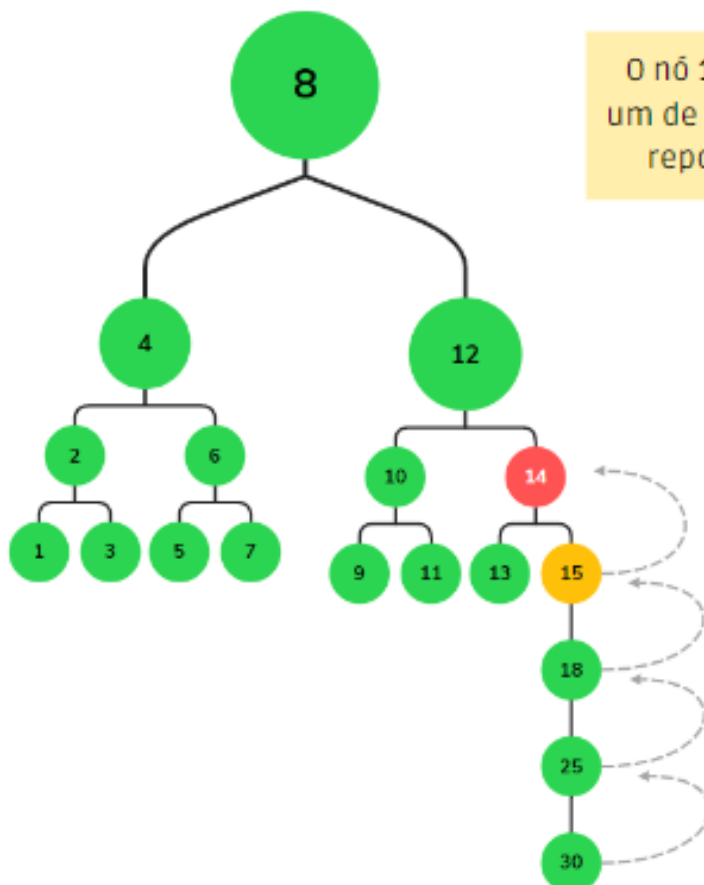
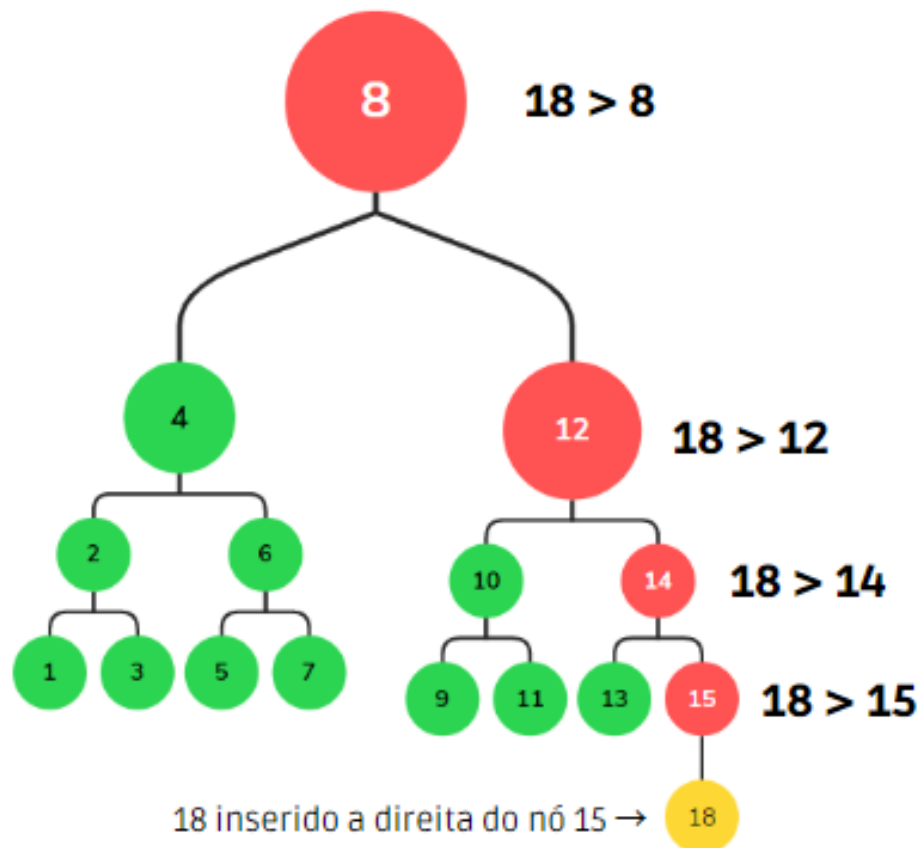
Para inserir um elemento N em uma árvore você deve seguir os passos:

- 1) Verifique se existe um nó raiz
  - a) Caso não exista, o elemento será a raiz
  - b) Caso exista verifique
    - i) Se o elemento N for menor que o nó inserir o elemento na **esquerda**
    - ii) Se o elemento N for maior que o nó inserir o elemento na **direita**

Este processo deve ser repetido recursivamente até que o elemento seja inserido na estrutura da árvore.



Vamos inserir os itens 18, 25, 30 na árvore, começando pelo item 18



O nó 14 informado será excluído, e um de seus filhos assumirá seu lugar reposicionando os demais nós

Se o nó não tem filhos, apenas remove.

Se o nó possui apenas um filho remove o pai e coloque o filho no lugar

O nó possui dois filhos, substituir pelo maior nó da árvore da esquerda, ou pelo menor nó da árvore a direita