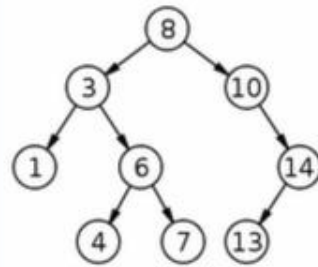


Árvore Binária em Python



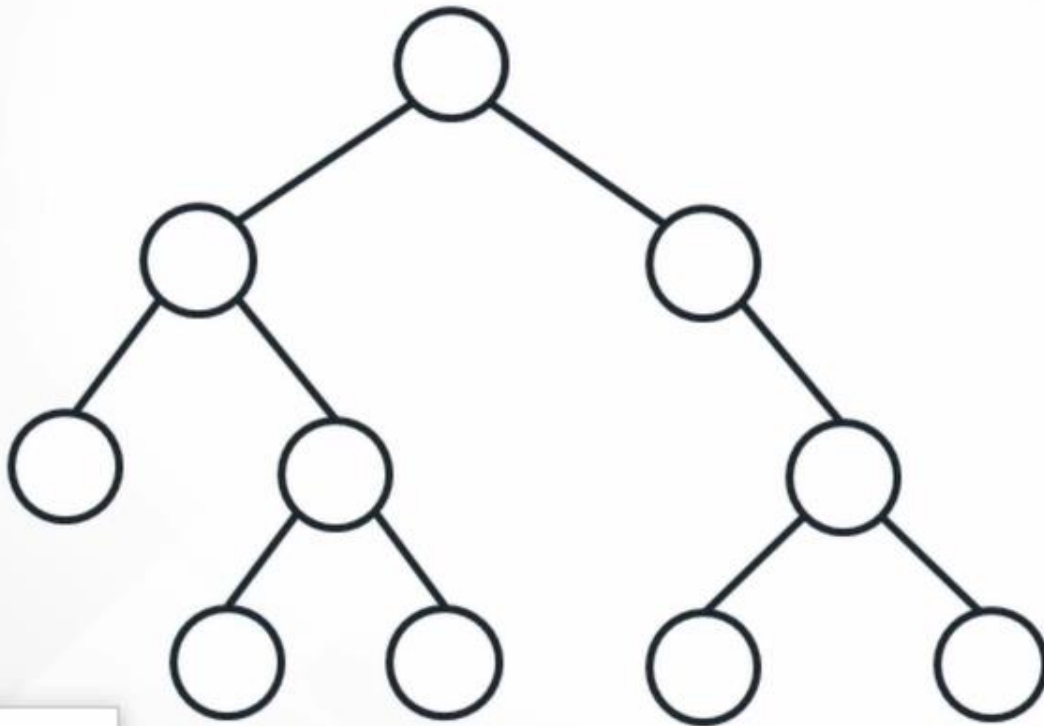
1. Conceitos de Árvores
2. Operações em Árvores



Árvores Binárias

Árvores Binárias

Árvores binárias são árvores nas quais cada nó pode ter no máximo **dois filhos**.



Árvores Binárias

Árvores Binárias em Python

Uma árvore é um conjunto de nós, e **cada nó é um objeto** com uma **chave** e uma **referência** aos seus dois filhos (esquerdo e direito).

```
class NoArvore:
    def __init__(self, chave = None, esquerda = None, direita = None):
        self.chave = chave
        self.esquerda = esquerda
        self.direita = direita
```



Árvores Binárias

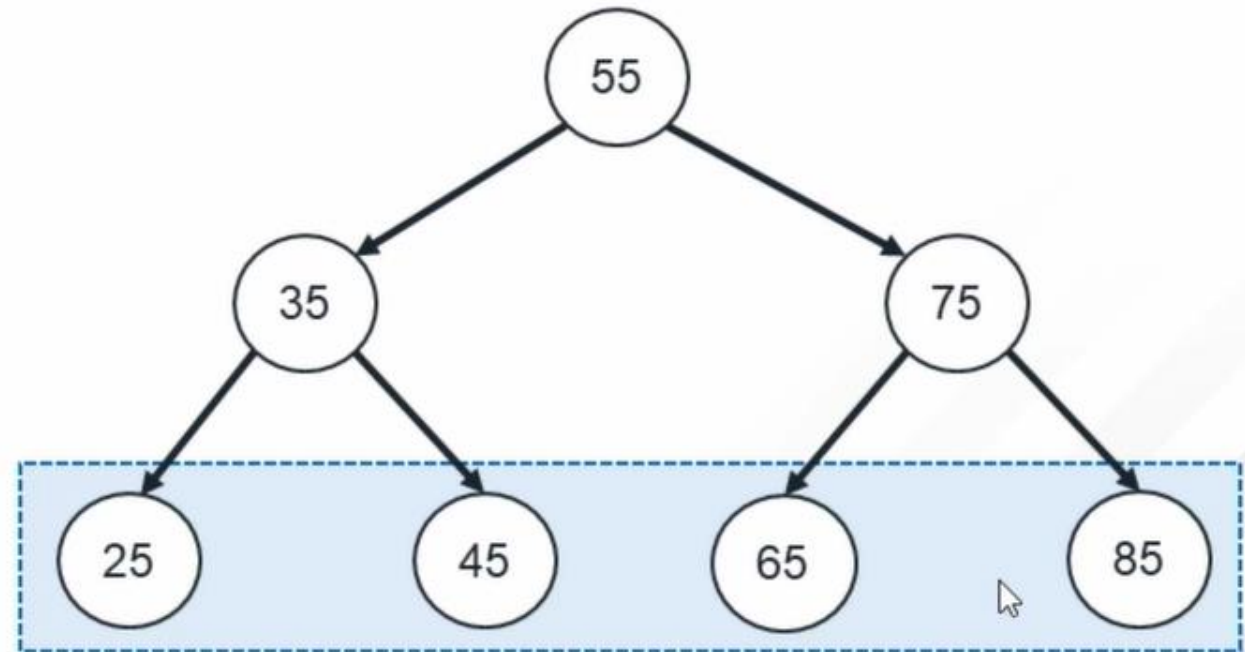
Árvores Binárias em Python

```
class NoArvore:  
    def __init__(self, chave = None, esquerda = None, direita = None):  
        self.chave = chave  
        self.esquerda = esquerda  
        self.direita = direita
```

```
if __name__ == '__main__':  
    raiz = NoArvore(55)
```

```
    raiz.esquerda = NoArvore(35)  
    raiz.direita = NoArvore(75)
```

```
    raiz.direita.esquerda = NoArvore(65)  
    raiz.direita.direita = NoArvore(85)  
    raiz.esquerda.esquerda = NoArvore(25)  
    raiz.esquerda.direita = NoArvore(45)
```



Árvores Binárias de Busca

Árvores Binárias de Busca

A **Árvore Binária de Busca** (do inglês **Binary Search Trees – BST**) são árvores cujo nós são organizados de acordo com as seguintes propriedades:

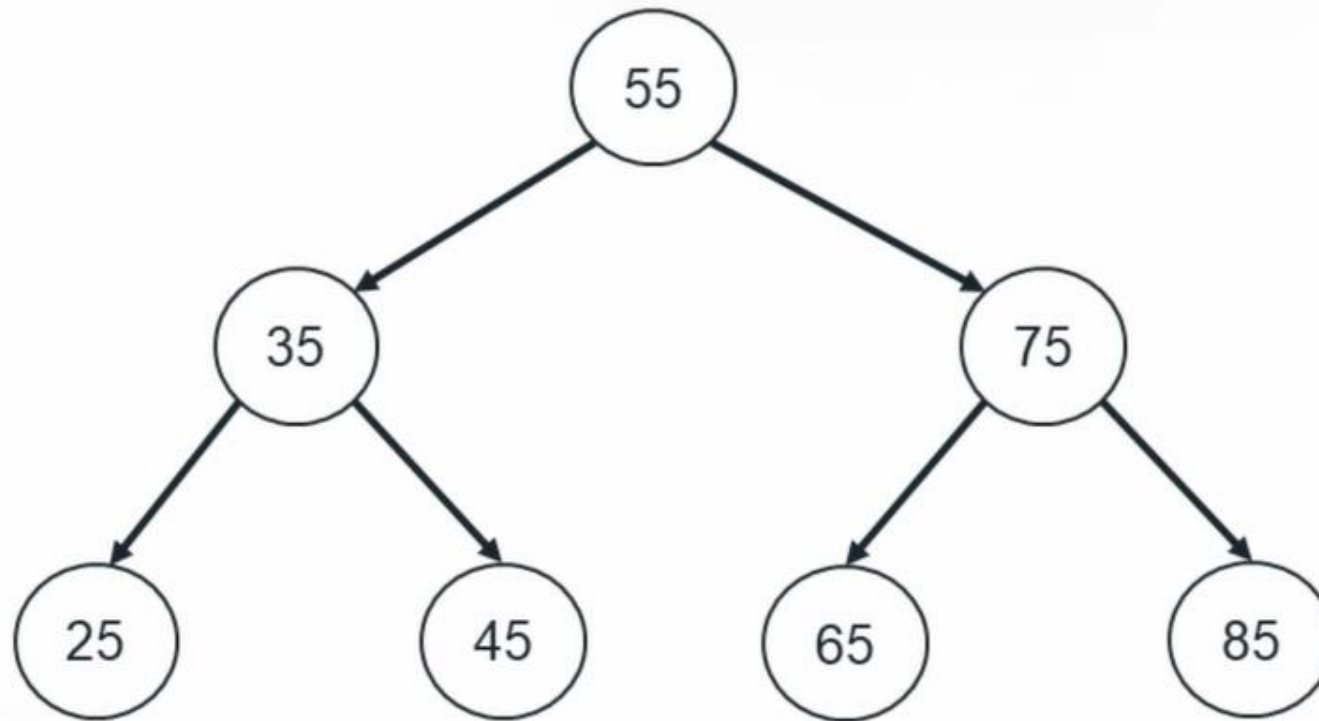
1. Dado um nó qualquer da árvore binária, todos os nós à **esquerda** dele são **menores** ou iguais a ele.
2. Dado um nó qualquer da árvore binária, todos os nós à **direita** dele são **maiores** ou iguais a ele.



Árvores Binárias de Busca

Árvores Binárias de Busca

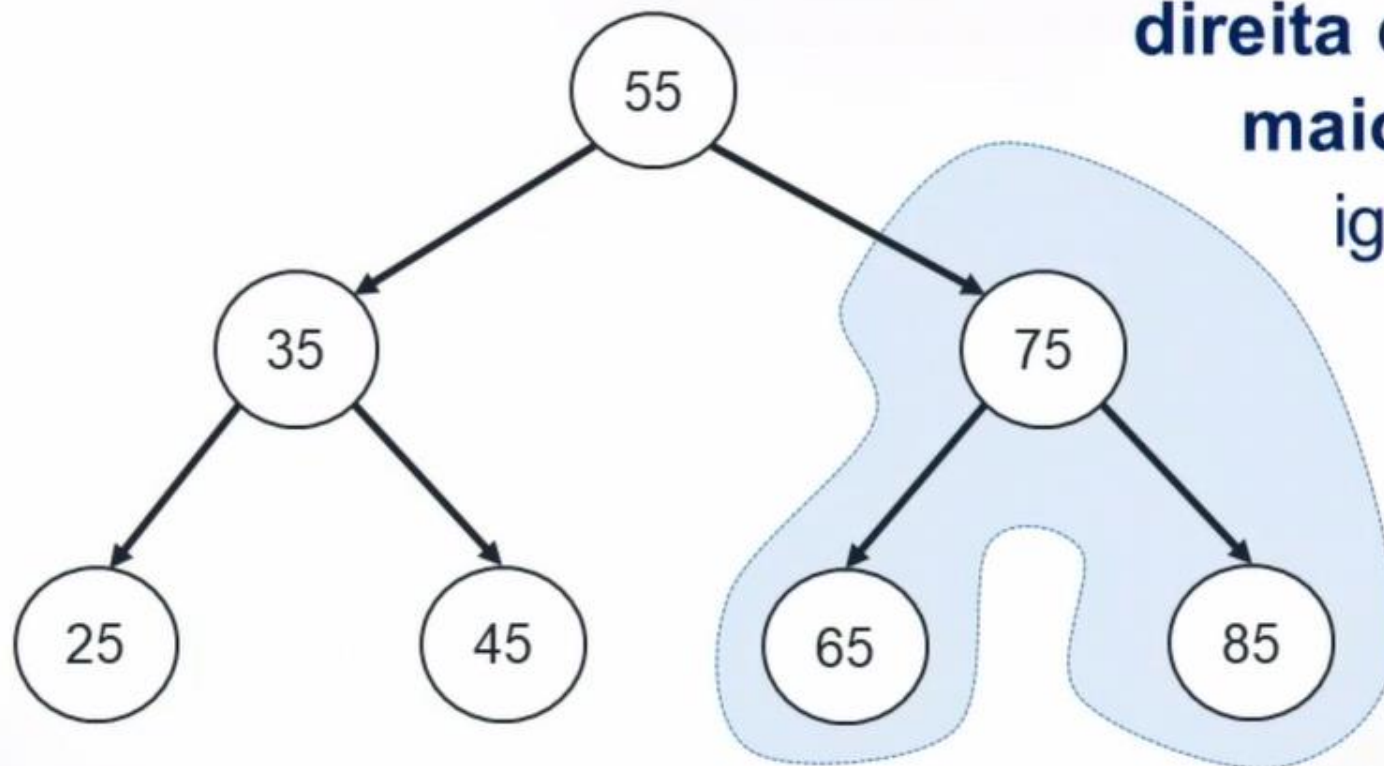
Exemplo de uma Árvore Binária de Busca:



Árvores Binárias de Busca

Árvores Binárias de Busca

Exemplo de uma Árvore Binária de Busca:



Todos os nós à **direita** da raiz são **maiores** ou iguais.