

Estrutura de Dados.

Nome: Adair Jose Ribeiro - 12109434

Questões sobre Árvore Binária de Pesquisa.

1. Encontre o menor elemento em uma Árvore Binária de Pesquisa (BST - Binary Search Tree)

Resposta: O menor elemento de uma BST é o nodo mais à esquerda na árvore.

```
def maximo(raiz): nodo = raiz while nodo.direita is not None: nodo = nodo.direita return nodo.chave
```

2. Encontre o maior elemento em uma Árvore Binária de Pesquisa (BST - Binary Search Tree)

Resposta: O maior elemento de uma BST é o nodo mais à direita na árvore.

```
def maximo(raiz): nodo = raiz while nodo.direita is not None: nodo = nodo.direita return nodo.chave
```

3. Verifique se duas árvores binárias são idênticas.

Resposta:

```
def identicas(a, b): # 1. As duas árvores são vazias. if a is None and b is None: return True # 2. Nenhuma das árvores é vazia. Precisamos compará-las. if a is not None and b is not None: return ((a.chave == b.chave) and identicas(a.esquerda, b.esquerda) and identicas(a.direita, b.direita)) # 3. Uma árvore é vazia mas a outra não. return False.
```

4. Calcule a altura de uma Árvore Binária.

Resposta: A altura de um nodo em uma árvore é o número de níveis desde o nodo em questão até a folha mais baixa da árvore. Quando falamos da altura de uma árvore, estamos nos referindo à altura do nodo raiz.

Por definição, a altura de uma árvore é a altura da raiz (que é 1), mais a altura da maior sub árvore. O código abaixo implementa essa ideia.

```
def altura(raiz):  
    if raiz is None:  
        return 0  
    return max(altura(raiz.esquerda), altura(raiz.direita)) + 1
```

5. Verifique se uma árvore binária é balanceada.

Resposta:

```
def balanceada(raiz): # Uma árvore binária vazia é balanceada. if raiz is None: return True  
altura_esq = altura(raiz.esquerda)  
altura_dir = altura(raiz.direita) # Alturas diferem em mais de uma unidade. if abs(altura_esq - altura_dir) > 1: return False  
return balanceada(raiz.esquerda) and balanceada(raiz.direita).
```