INSTITUTO FEDERAL DE ALAGOAS
CURSO BACHARELADO EM SISTEMAS DE INFORMAÇÃO
Estruturas de Dados
Prof. Ricardo Rubens
ricardo (arroba) ifal.edu.br

Lista de Exercícios 11 (Árvores/P1)

1. Explique por que uma árvore é uma estrutura de dados não-linear.

2. Qual a diferença entre um nó interno e um nó folha em uma árvore?

3. Como a hierarquia de um diretório de arquivos pode ser representada como uma árvore?

4. O que é um caminho em uma árvore e como ele é formado?

5. Por que árvores são úteis na representação de estruturas hierárquicas?

6. Desenhe a estrutura de árvore resultante do seguinte conjunto de chamadas de função de árvore:

```
r = BinaryTree('a')
insert_left(r, 'b')
insert_left(r, 'c')
insert_right(r, 'd')
insert_left(r, 'e')
set_root_val(r, 'g')
insert_left(r, 'h')
```

7. Uma árvore binária encadeada mantém uma referência de cada nó para seu sucessor. Modifique o código de uma árvore de pesquisa binária para torná-la encadeada e, em seguida, escreva um método de travessia inorder não recursivo para a árvore de pesquisa binária encadeada.

8. Escreva um código python que construa uma árvore binária usando listas de listas para representar a seguinte estrutura:

```
['a', ['b', ['c', [], [d]], ['e', [f], []]], ['g', ['h', [i], []], []]]
```

9. Implemente a função `eh_folha(tree)` que retorna `True` se a árvore contém apenas um nó (ou seja, a raiz não tem filhos), e `False` caso contrário.

```python
# Exemplo:
print(eh_folha(['a', [], []]))  # True
print(eh_folha(['a', ['b', [], []], []]))  # False
```

10. Implemente a função `tem_filho(tree)` que retorna `True` se a árvore contém pelo menos um filho, e `False` caso contrário.

```python
# Exemplo:
print(tem_filho(['a', [], []]))  # False
print(tem_filho(['a', ['b', [], []], []]))  # True
```

11. Implemente uma função `altura(tree)` que retorna a **altura** de uma árvore binária representada como listas de listas.

```python
def altura(tree):
    ...
```

12. Implemente uma função `contar_nos(tree)` que retorna a **quantidade total de nós** na árvore.

```python
def contar_nos(tree):
    ...
```