

# Teoria de Grafos

## Trabalho 6

### Enunciado

O trabalho consiste em implementar uma série de funções que serão colocadas nos módulos `Arvore.hs`, a ser criado por você, e `Grafo.hs`, do trabalho anterior, e cujos enunciados serão dados na sequência. Importe quaisquer módulos de trabalhos anteriores que precisar.

Crie um arquivo de testes `Teste6.hs` que importe o módulo `Arvore.hs` e também os módulos `Grafo.hs` e `GrafosEspeciais.hs`, de trabalhos anteriores, para testar cada uma dessas funções em pelo menos três árvores diferentes e que não sejam apenas variações do mesmo árvore.

**Não** use biblioteca alguma que implemente diretamente as funções pedidas.

A função a seguir deve ser acrescentada ao módulo `Grafo.hs` criado em trabalhos anteriores.

**Ex. 1** `éSubgrafoGerador g h` devolve `True` se o grafo `h` é um subgrafo gerador do grafo `g` ou `False` em caso contrário.

As funções a seguir devem ser implementadas no novo módulo `Arvore.hs`.

**Ex. 2** `éÁrvore g`, devolve `True` se o grafo `g` é uma árvore ou `False` em caso contrário.

**Ex. 3** `éEstrela g` devolve `True` se o grafo `g` é estrela ou `False` em caso contrário.

**Ex. 4** `éFloresta g` devolve `True` se o grafo `g` é floresta ou `False` em caso contrário.

**Ex. 5** `folhas a`, devolve uma lista de vértices que são folhas na árvore `a`.

**Ex. 6** `internos a`, devolve uma lista com os vértices internos da árvore `a`.

**Ex. 7** `centro a`, devolve uma lista contendo os vértices no centro da árvore `a`. Use o algoritmo específico para árvores.

**Ex. 8** `éAncestral a r v w` devolve `True` se o vértice `v` é ancestral do vértice `w` na árvore `a` enraizada no vértice `r`.

**Ex. 9** `éDescendente a r v w` devolve `True` se o vértice `w` é descendente do vértice `v` na árvore `a` enraizada no vértice `r`.

**Ex. 10** `éIrmão a r v w` devolve `True` se o vértice `v` é irmão do vértice `w` na árvore `a` enraizada no vértice `r`.

**Ex. 11** `ancestrais a r v` devolve uma lista contendo todos os vértices *ancestrais próprios* do vértice `v` na árvore `a` enraizada no vértice `r`.

- Ex. 12** **descendentes a r v** devolve uma lista contendo todos os vértices *descendentes próprios* do vértice **v** na árvore **a** enraizada no vértice **r**.
- Ex. 13** **nível a r v** devolve o nível do vértice **v** na árvore **a** enraizada no vértice **r**.
- Ex. 14** **altura a r** devolve a altura da árvore **a** enraizada no vértice **r**.
- Ex. 15** **parcial a r s** devolve uma subárvore parcial da árvore **a** enraizada no vértice **r** após a remoção dos vértices na lista **s**.
- Ex. 16** **éBinária a r** devolve **True** se a árvore **a** enraizada no vértice **r** é uma árvore binária ou **False** em caso contrário.
- Ex. 17** **éBinEstrita a r** devolve **True** se a árvore **a** enraizada no vértice **r** é uma árvore estritamente binária ou **False** em caso contrário.
- Ex. 18** **éBinCompleta a r** devolve **True** se a árvore **a** enraizada no vértice **r** é uma árvore binária completa ou **False** em caso contrário.
- Ex. 19** **éÁrvoreM a r m** devolve **True** se a árvore **a** enraizada no vértice **r** é uma árvore **m**-ária ou **False** em caso contrário.
- Ex. 20** **éMcheia a r m** devolve **True** se a árvore **a** enraizada no vértice **r** é uma árvore **m**-ária cheia ou **False** em caso contrário.
- Ex. 21** **éGeradora g a** devolve **True** se a árvore **a** é uma árvore geradora do grafo **g** ou **False** em caso contrário.
- Ex. 22** **geradora g** devolve uma árvore geradora do grafo **g**.
- Ex. 23** **geradoras g** devolve uma lista contendo *todas* as árvores geradoras do grafo **g**.