

Lista de exercícios – (3 pontos)
Entrega do trabalho no SGA (PDF): 15/05/2018

Atenção: não serão aceitas cópias de trabalhos/questões. Vamos utilizar o critério de **honestidade acadêmica**. Para desenhar as árvores, utilize o programa yEd Graph Editor disponível em http://www.yworks.com/en/products_yed_download.html.

OBS: entregue seu trabalho em formato PDF. Trabalhos em outros formatos não serão aceitos.

1 – Para cada sequência abaixo pede-se:

- a) Desenhe a ABP
 - a. Monte uma tabela com a altura e profundidade de cada nó
 - b. Qual a ordem de impressão dos nós nos percursos pré-ordem e pós-ordem
 - b) Desenhe a árvore AVL
 - a. Monte uma tabela com a altura e profundidade de cada nó
 - b. Qual a ordem de impressão dos nós nos percursos pré-ordem e pós-ordem
- i. 1, 2, 7, 19, 4, 15, 8, 6, 10 e 13
 - ii. 32, 63, 41, 23, 16, 27, 31, 50, 1, 5
 - iii. 35, 30, 32, 67, 79, 15, 18, 26, 29, 70, 80, 19, 21, 8, 16, 1, 126 e 74
 - iv. Maria, José, João, Ana, Tiago, Nico, Taís, Davi, Tião, André e Savio

2 – Considere a árvore binária gerada pela inserção da seguinte sequência de chaves (sequência iii do exercício acima): 35, 30, 32, 67, 79, 15, 18, 26, 29, 70, 80, 19, 21, 8, 16, 1, 126 e 74. Pede-se:

- a) Desenhe a árvore resultante das remoções dos nós 18, 35, 126 e 67 sucessivamente.

3 – Crie um método recursivo que percorra a árvore em pré-ordem e imprima quantos filhos cada nó tem.

4 – Crie um método recursivo que percorra a árvore em ordem crescente e imprima quantos filhos cada nó tem.

5 – Crie um método recursivo que percorra a árvore em pós-ordem e imprima quantos filhos cada nó tem.

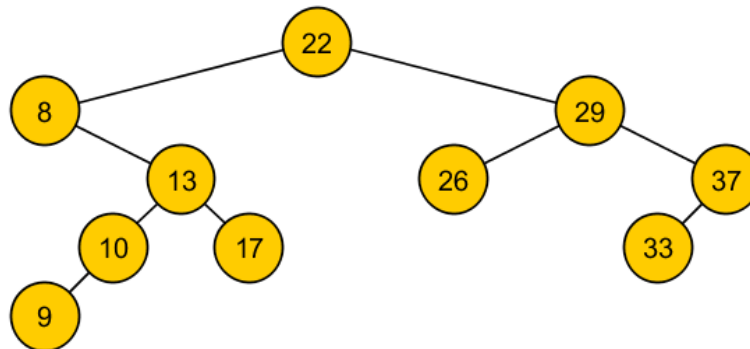
6 – Crie um método recursivo que percorra a árvore em pré-ordem e imprima, para cada nó, quais são os filhos da esquerda e direita.

7 – Crie um método recursivo que percorra a árvore em ordem crescente e imprima, para cada nó, quais são os filhos da esquerda e direita.

8 – Crie um método recursivo que percorra a árvore em pós-ordem e imprima, para cada nó, quais são os filhos da esquerda e direita.

9 – Crie um método recursivo que retorne a altura de um nó passado por parâmetro.

- 10 – Crie um método recursivo que retorne a profundidade de um nó passado por parâmetro.
- 11 – Crie um método que calcule e retorne o Fator de Balanceamento de um nó passado por parâmetro.
- 12 – Crie o método recursivo **private void imprimeInfoNo** para que ele imprima, para cada nó da árvore binária, o seu nível, a quantidade de filhos do nó (0, 1 ou 2) e quais são os filhos (se existirem). O caminhamento utilizado deve ser o pré-ordem. O layout de sua saída deve ser IDÊNTICO ao do exemplo abaixo:



```
* Nó 22 - nível 1 - 2 filho(s) : 8 e 29
* Nó 8 - nível 2 - 1 filho(s) : 13
* Nó 13 - nível 3 - 2 filho(s) : 10 e 17
* Nó 10 - nível 4 - 1 filho(s) : 9
* Nó 9 - nível 5 - 0 filho(s)
* ...
* Nó 33 - nível 4 - 0 filho(s)
```