**Orientações:**

* Leia atentamente as questões antes de respondê-las. A leitura atenta e a interpretação correta contarão muito para um bom resultado de sua avaliação. Obviamente, a interpretação faz parte da avaliação.
* Tenha seu estilo próprio, sem cópias.
* As funções devem ser implementadas de forma **otimizada**. Ou seja, onde for possível utilizar a vantagem de a árvore ser binária de busca, isso será desejado.
* **prints ou returns** usados desnecessariamente ou de forma errônea serão considerados negativamente na avaliação (de igual forma se um None aparecer porque você mandou imprimir uma função que não retorna nada).
* Você pode utilizar **somente seus programas, slides e textos que tiver baixado** como objetos de consulta. Não poderá haver acesso a Internet durante a realização da avaliação.
* No arquivo de resposta apresente **somente as funções necessárias** para a execução do código. Siga os nomes utilizados em aula.
* Compacte seus arquivos em um único arquivo rar ou zip com seu nome completo. Exemplo: Fabiano\_Fagundes.rar. Suba este arquivo na atividade correspondente no Classroom.
* Se você não atender ao que foi pedido você perderá 20% do resultado da sua avaliação.
* Se você não ler nada disso e perguntar algo que está aí acima você perderá 10% do resultado da sua avaliação.
* Não tirarei dúvidas.

**Questão 1 (valor 2,0):2,0**

Implemente, para uma Árvore Binária de Busca (ABB), a função nos\_com\_descendente\_par(self) que tem como tarefa imprimir os valores dos nós que tenham ao menos um descendente com valor par.

**Como ver se um valor é par:** o operador % retorna o resto da divisão de um número. Quando usamos o número 2 como divisor, se o resto for zero, o número é par; caso contrário, será ímpar.

**Questão 2 (valor 2,0):2,0**

Implemente, para uma Árvore Binária de Busca (ABB), a função tem\_descendente\_par(self, valor) que retorna True se o nó que contém o valor passado como argumento tiver ao menos um descendente par e False caso contrário.

**Questão 3 (valor 1,5):1,5**

Implemente, para uma Árvore Binária de Busca (ABB), a função imprime\_maior\_filho(self, valor) que imprime o maior valor entre os descendentes do nó que contém o valor passado como argumento.