

**Escola Superior de Gestão e Tecnologia de Santarém****Licenciatura em Informática****2º Ano/1º Semestre****2022/2023****Algoritmos e Estruturas de Dados****Ficha de trabalho n.º 8****Análise de complexidade e notação Big-O**

**ATENÇÃO:** NOS PROGRAMAS SEGUINTE NÃO UTILIZE AS CLASSES JÁ PROGRAMADAS DA PACKAGE JAVA.UTIL OU A COLLECTIONS DO JAVA.

1. Crie uma classe que implemente a representação de um Livro com ISBN, autor e nome e que se possa auto imprimir de forma apresentável.
2. Crie uma classe que implemente uma árvore binária de livros.
3. Crie uma classe que implemente uma lista de livros.
4. Crie um método na classe principal que crie uma árvore binária de livros perguntados ao utilizador até que este diga que não quer introduzir mais indicando para isso um nome vazio.

As regras para construir a árvore devem ser as seguintes:

1. O primeiro livro é a raiz
2. Os livros restantes são distribuídos da seguinte forma:
  - a. Se o livro a introduzir tiver um nome alfabeticamente menor que o do livro no nó corrente, o livro a introduzir deve ser colocada na árvore esquerda desse nó, se esta não existir, é criado um nó para esse livro e colocado como nó esquerdo do nó corrente.
  - b. Caso contrário, deve ser colocado na árvore direita, e se esta não existir, deve ser criado um nó para esse livro e colocado com nó direito do nó corrente.

O método para introduzir um novo livro na árvore de livros deve fazer parte da definição da classe da árvore de livros.

5. Analise a complexidade de 4) (método para inserir) e descreva-a em notação Big-O. Consegue melhor?
6. Pense num algoritmo de pesquisa de ordem  $O(n)$  na lista de 3). Trata-se de encontrar um livro dado o seu nome, e depois imprimir.

7. Pense num algoritmo melhor e descreva-o, programe-o e diga qual a sua complexidade utilizando notação Big-O.
8. Descreva um algoritmo para encontrar um livro na árvore de 4) dado o seu nome, e diga qual a sua complexidade em notação Big-O.