

### Avaliação 03

#### Descritivo de Implementação

O presente documento tem por objetivo servir de guia para a elaboração dos trabalhos da **terceira** unidade da componente curricular Estruturas de Dados Básicas II. Ao fim desta unidade, é esperado que o aluno tenha conhecimento sobre a teoria relativa ao conteúdo abordado, e consiga replicá-la em um ambiente computacional.

Como ficou definido na aula inicial, as atividades avaliativas irão constar de implementações computacionais dos pressupostos teóricos vistos em sala de aula. Tais implementações deverão seguir algumas regras básicas.

- (a) As implementações deverão ser feitas em grupo de até 5 participantes (discentes matriculados na turma);
- (b) A linguagem a ser adotada é de livre escolha. Contudo, as implementações devem abranger a teoria vista integralmente. O uso de bibliotecas auxiliares é permitido, porém, não para as atividades fim dos algoritmos. Em outras palavras, as estruturas de dados e pseudocódigos devem ser implementados em sua integralidade;
- (c) Apesar de ser uma atividade em grupo, a avaliação ocorrerá de forma individual, de forma que todos devem ter conhecimento dos códigos e funções implementadas pelo grupo.
- (d) Não custa lembrar que a presença física nas aulas também é objeto de avaliação, contribuindo para a construção da nota individual. Então, evitem faltar.
- (e) A data para entrega (virtual) é o dia **28 de janeiro de 2025**.

Deverá ser entregue:

- i. **O código utilizado, para inspeção:** Os códigos deverão estar comentados, em português, detalhando o funcionamento de cada função/procedimento, incluindo o formato da entrada e qual a saída esperada (pode ser GitHub).
- ii. **Um vídeo curto**, explicando o funcionamento/estrutura geral do programa, IDE utilizada e os resultados encontrados ao executar os algoritmos.
- iii. **Uma lista de atividades desenvolvidas por integrante**, detalhando a participação efetiva em cada implementação, seja em sua concepção, implementação, revisão ou teste.

Para a avaliação, deverá ser considerado o seguinte:

## 1. Implementação Computacional (Teoria)

**1.0.** Descreva o ambiente computacional utilizado (*Software* e *Hardware*).

**1.1.** Faça um algoritmo em forma de fluxograma para a função de exclusão de uma informação em uma árvore B. (deve ser entregue em formato PDF)

## 2. Implementação Computacional (Aplicação, Árvore B)

**2.0.** Implemente uma árvore B para armazenar e organizar informações de um sistema de controle de estoque. Cada nó da árvore conterá itens de estoque com as seguintes informações:

- ID do item (chave de busca - tipo inteiro)
- Nome do item (descrição - tipo string)
- Quantidade em estoque (tipo inteiro)

Os dados serão inseridos, pesquisados e excluídos usando a estrutura de árvore B. Os dados a serem inseridos estão no arquivo *dadosB.txt* disponível no SIGAA.

### 2.1. Criação da Estrutura:

- Implemente uma árvore B com grau mínimo  $t$ . (*Considerar as opções  $t=2$  e  $t=3$* )
- Cada nó armazenará até  $2t$  chaves e terá até  $2t+1$  filhos.

### 2.2. Operações para Implementar:

**Inserção:** Adicione os dados fornecidos, gerando uma árvore B balanceada. O usuário deve ser capaz de inserir novos itens no sistema.

**Busca:** Permita pesquisar um item pelo ID (ex., buscar pelo ID 109 deve retornar {109, "SSD", 30}).

**Exclusão:** Remova um item da árvore através do ID (ex., excluir o ID 140).

### 2.3. Visualização:

- Imprima os IDS da árvore por níveis, separando as páginas por colchetes. Exemplo de saída com duas páginas no nível 1: Nível 1: [50, 110] [150, 203]

Ou gere um arquivo ".dot" para visualizar a estrutura usando o *Graphviz*.