

**ATIVIDADE 3 - ESOFT - ESTRUTURA DE DADOS II - 52/2023****Período:**12/06/2023 08:00 a 30/06/2023 23:59 (Horário de Brasília)**Status:**ABERTO**Nota máxima:**0,50**Gabarito:**Gabarito não está liberado!**Nota obtida:****1ª QUESTÃO**

Caro estudante,

Temos por certo que os desafios sempre contribuem na aquisição de conhecimentos e competências desejadas. Assim, faz-se necessário relacionar o que se aprende com situações reais que podem ser encontradas no cotidiano.

Nesta atividade, você é convidado a realizar uma atividade para verificar como a disciplina em questão pode contribuir na sua experiência e formação profissional. Por este motivo, nesta atividade discursiva, você é instigado a solucionar um problema voltado para sua área de formação.

Atente-se para as informações e realize um ótimo trabalho!

-----

A maioria das linguagens de programação possuem estruturas de dados lineares implementadas nativamente em suas bibliotecas padrão. Por exemplo, linguagens como o C, C++, Java, C#, JavaScript, Python, dentre tantas outras, possuem algum tipo de arranjo linear (array) que, por vezes, é chamado de vetor, em outros momentos é chamado de lista, e assim por diante. Todavia, raras vezes tais linguagens apresentam uma implementação explícita de árvores binárias. Isso se deve ao fato de que, para desenvolvedores de alto nível, não é preciso apresentar a estrutura de uma árvore em sua complexidade.

Estruturas de dados como dicionários, listas ordenadas, mapeamentos e outras estruturas indexadas, muitas vezes, por debaixo dos panos, estão utilizando árvores para que as operações de inserção, consulta, alteração e remoção de elementos em tais estruturas sejam feitas de forma eficiente. É o caso da Linguagem C++ que, em sua implementação padrão, entrega uma estrutura de dados chamada "map" que, internamente, implementa uma árvore de busca binária.

Fonte: OLIVEIRA, P. M. de; PEREIRA, R. de L. **Estruturas de Dados II**. Maringá: UniCesumar, 2019.

Assim, coloque-se no papel de um(a) desenvolvedor(a) de infraestrutura, que precisa demonstrar que sabe operar uma árvore de busca binária balanceada, utilizando o algoritmo de balanceamento AVL. Leve em consideração a seguinte ordem de inserção de elementos, e leve em conta que "elementos menores vão à esquerda, e elementos maiores ou iguais vão à direita":

**10, 05, 20, 40, 50, 04, 03, 08, 07, 15, 18**

Você pode desenhar a árvore AVL balanceada resultante em uma folha de papel, bater uma foto e enviar como resposta. Alternativamente, você pode ilustrar sua árvore em algum software de ilustração como o Microsoft Paint ou o Microsoft PowerPoint, e nos encaminhar o print em formato de imagem.

Bons estudos!

**ALTERNATIVAS**

Arquivo enviado : AE03\_estruturaDadosII\_RA211509305.pdf