

Trabalho 01: Análise Comparativa de Estruturas de Dados

#Descrição Geral:

O trabalho tem como objetivo comparar diferentes estruturas de dados: i) arrays lineares, ii) árvores de busca e iii) tabelas hash, em operações de inserção e busca, avaliando múltiplas métricas de desempenho. O foco principal é desenvolver uma análise crítica sobre a eficiência estrutural e o custo computacional de cada abordagem.

#Objetivos de Aprendizagem

- Implementar arrays, árvores de busca binária e tabelas hash em Python ou C (sem utilizar bibliotecas que já forneçam essas estruturas).
- Comparar o desempenho das diferentes estruturas e suas parametrizações, considerando diferentes volumes de dados, tanto no armazenamento dos dados quanto nas operações de busca, permitindo assim uma análise crítica de eficiência e custo computacional.
- Avaliar o consumo de recursos, incluindo número de iterações, utilização de memória e carga de CPU, durante operações de inserção e busca.
- Desenvolver uma análise crítica das vantagens, limitações e trade-offs de cada estrutura de dados, relacionando conceitos teóricos com resultados experimentais obtidos.
- Produzir uma documentação técnica em formato de artigo científico, seguindo normas acadêmicas de escrita e apresentação de resultados.

#Requisitos:

Dados

- Criar registros fictícios com diferentes volumes: $N = \{10.000, 50.000, 100.000, \dots\}$
- Cada registro deve conter, por exemplo: Matrícula (9 dígitos), Nome, Salário, Código do Setor, Outros campos relevantes conforme necessidade do experimento

Estruturas de Dados

- **Array Linear**
- **Árvore de busca binária (BST)**
 - Avaliar impacto com e sem balanceamento
- **Tabela Hash**
 - Avaliar três distintas funções hash
 - Tratamento de colisões ($M = \{100, 1000, 5000\}$)
 - Avaliar colisões, *load factor*

#Relatório de Análise Crítica

- Avaliar o consumo de recursos, incluindo memória, CPU e outros indicadores relevantes (i.e iterações). Recomenda-se o uso de bibliotecas e ferramentas, como **tracemalloc** e **psutil**, para coletar métricas detalhadas de consumo de memória e utilização de CPU durante a execução dos experimentos.
- Desenvolver uma análise crítica das vantagens e limitações de cada estrutura de dados, incluindo a análise assintótica (notação Big-O) das operações estudadas.
- Realizar a análise considerando tanto a organização interna dos dados quanto a performance em operações de busca.
- Executar os experimentos em computador local, garantindo um ambiente controlado e estável, em contraposição a máquinas virtuais ou notebooks colaborativos (Colab ou similares).
- Para cada cenário de teste (por exemplo, diferentes volumes de dados ou parametrizações de tabela hash), deve-se realizar ao menos cinco rodadas independentes.
 - Motivação: múltiplas execuções permitem reduzir o impacto de variabilidades, concorrências e flutuações do sistema operacional, fornecendo médias e desvios padrões mais confiáveis para análise comparativa.

#Normas e Artefatos:

- Código-fonte: .py ou .c (Não utilizar jupyter).
- Relatório no padrão de artigo científico (ieee – duas colunas), com no máximo 6 páginas.
- Atividade em Grupo formado no ambiente virtual (Canva)
 - Entregar os artefatos em formato zip, na atividade aberta no Canva

#Rúbricas de Avaliação:

Tabela 1 – Implementação e Funcionamento das Estruturas de Dados (35 pontos)(*)

<u>Estrutura</u>	<u>Excelente</u>	<u>Bom</u>	<u>Regular</u>	<u>Insuficiente</u>	<u>Peso</u>
<u>Array Linear</u>	Implementação correta, eficiente e funcional para todos os volumes de dados.	Funcional, com pequenas inconsistências ou limitações.	Parcialmente funcional, com erros em alguns casos.	Ausente ou não funcional.	10
<u>Árvore de Busca Binária (BST)</u>	Inserção, busca e balanceamento corretos para todos os casos.	Funcional, mas balanceamento ou buscas incompletos.	Parcialmente funcional, erros em inserção ou busca.	Ausente ou não funcional.	10
<u>Tabela Hash</u>	Implementação completa com múltiplas funções hash, tratamento de colisões e variação de M corretamente aplicada.	Funcional, mas apenas uma função hash ou tratamento parcial de colisões.	Implementação incompleta, falhas na função hash ou colisões.	Ausente ou não funcional.	10
<u>Código e documentação</u>	Código bem organizado, comentado, instruções de execução claras.	Código funcional, mas documentação parcial ou comentários insuficientes.	Código confuso, pouco comentado, execução difícil.	Código ausente ou não executável.	5

Tabela 2 – Análise de Desempenho e Métricas (30 pontos)

<u>Subcritério</u>	<u>Excelente</u>	<u>Bom</u>	<u>Regular</u>	<u>Insuficiente</u>	<u>Peso</u>
<u>Coleta de métricas (CPU, memória, iterações)</u>	Todas métricas coletadas corretamente; resultados claros e organizados.	Métricas coletadas, mas com lacunas ou apresentação pouco clara.	Métricas incompletas ou inconsistentes ; análise limitada.	Métricas ausentes ou incorretas.	10
<u>Experimentos e replicabilidade</u>	Cinco ou mais rodadas independentes; média e desvio padrão corretos.	Rodadas realizadas, mas poucas ou análise parcial.	Rodadas limitadas ou sem análise estatística adequada.	Experimentos ausentes ou não replicáveis.	10
<u>Avaliação de diferentes populações de dados e parametrizações</u>	Análise clara e detalhada para múltiplos N, M, funções hash e balanceamento; impacto das parametrizações discutido.	Avaliação realizada, mas cobertura parcial das populações ou parametrizações.	Apenas alguns cenários considerados; análise limitada.	Cenários de teste ausentes ou análise inexistente.	10

Tabela 3 – Relatório Científico (35 pontos)(*)

<u>Subcritério</u>	<u>Excelente</u>	<u>Bom</u>	<u>Regular</u>	<u>Insuficiente</u>	<u>Peso</u>
<u>Formatação e normas acadêmicas</u>	Segue padrão IEEE, duas colunas, referências corretas, até 6 páginas.	Pequenas inconsistências na formatação ou referências.	Formatação irregular, referências incompletas ou confusas.	Não segue normas acadêmicas, referências ausentes.	5

<u>Clareza e argumentação técnica</u>	Discussão crítica clara; resultados apresentados com tabelas, gráficos e ferramentas analíticas; relaciona teoria e prática.	Discussão adequada, mas uso de tabelas/gráficos parcial ou análise superficial.	Argumentação limitada, poucos recursos visuais/análise gráfica, pouca relação teoria-resultados.	Discussão ausente ou incoerente; sem uso de ferramentas de análise.	10
<u>Análise comparativa</u>	Comparação detalhada entre estruturas de dados, relacionando consumo de recursos, Big-O e trade-offs.	Comparação adequada, mas superficial ou incompleta.	Comparação limitada, sem relação clara com teoria.	Comparação ausente ou incorreta.	20

(*) A NÃO ENTREGA e/ou ENTREGA INCOMPLETA dos artefatos: I) Código-fonte (.py ou .c) e/OU II) Relatório técnico (padrão IEEE, duas colunas, até 6 páginas), invalida automaticamente todos os demais critérios da avaliação, recebendo nota zero nos subcritérios relacionados à implementação, análise de desempenho e relatório técnico.