



UNIVERSIDADE PRESBITERIANA MACKENZIE  
Decanato Acadêmico



<b>Unidade Universitária:</b> Faculdade de Computação e Informática		
<b>Curso:</b> CIÊNCIA DA COMPUTAÇÃO		<b>Núcleo Temático:</b> Algoritmos e Programação
<b>Disciplina:</b> Algoritmos e Programação II		<b>Código da Disciplina:</b>
<b>Professor(es):</b> VALÉRIA FARINAZZO MARTINS	<b>DRT:</b> 1126910	<b>Etapas:</b> 2
<b>Carga horária:</b> 04h/a	( 02 ) Teórica ( 02 ) Prática	<b>Semestre Letivo:</b> 1º sem. 2020
<b>Ementa:</b> Estudo das estruturas de dados não-lineares (árvores binárias, heaps, árvores B, tabelas de hashing, mapas e dicionários) e suas aplicações. Análise assintótica de operações em estruturas de dados não lineares. Prática de implementação de estruturas de dados não-lineares com linguagem orientada a objetos.		
<b>Objetivos:</b>		
<b>Fatos e Conceitos</b>	<b>Procedimentos e Habilidades</b>	<b>Atitudes, Normas e Valores</b>
<ul style="list-style-type: none"><li>• <b>Continuar o estudo</b> de Tipos Abstratos de Dados (TAD), agora com enfoque em estruturas de dados não-lineares (árvores, tabelas de hashing, mapas, dicionários e conjuntos).</li><li>• <b>Praticar</b> com implementação e aplicação de estruturas de dados não lineares em C++ para <b>resolução eficiente</b> de problemas.</li></ul>	<ul style="list-style-type: none"><li>• Identificar a estrutura de dados adequada para a resolução de problemas.</li><li>• Saber o correto uso das estruturas de dados estudadas e sua eficiência em termos de tempo e espaço.</li></ul>	<ul style="list-style-type: none"><li>• Ampliar a habilidade de solução de problemas.</li><li>• Estabelecer um caráter crítico na resolução de problemas de forma eficiente.</li></ul>



**Conteúdo Programático:**

0. Apresentação do Plano de Ensino e dos pré-Requisitos

0.1. Revisão das estruturas de dados lineares

1. Árvores

1.1. Conceito

2. Árvores Binárias

2.1. Conceito

2.2. Algoritmos de Inserção, Remoção e Busca

2.3. Desempenho

3. Árvores Binárias de Busca

3.1. Conceito

3.2. Algoritmos de Inserção, Remoção e Busca

3.3. Desempenho

4. Árvores AVL

4.1. Conceito

4.2. Algoritmos de Balanceamento, Inserção e Remoção

4.3. Desempenho

5. Árvores B e B+

5.1. Conceito

5.2. Algoritmos de Inserção, Remoção e Busca

5.3. Desempenho

6. Hash Table

6.1 Conceitos

6.2 Algoritmos: endereçamento direto, lista encadeada, hashing com encadeamento, hashing com sondagem linear

7. Mapas



7.1. Conceito

7.2. Algoritmos de Inserção e Remoção

**Metodologia:**

- Aulas expositivas.
- Aulas práticas em laboratórios.
- Utilização do ambiente Mackenzie Virtual.

**Critério de Avaliação:**

**Nota 1 (N1)** composta de:

- Prova parcial 1 (**P1**) escrita, individual e sem consulta (70%) – NOTA A (TIA)
- Prova de Laboratório 1a (**Lab1a**) (20%) – NOTA B (TIA)
- Exercícios de Laboratório 1b (**Lab1b**) (10%) – NOTA C (TIA)

**Nota 2 (N2)** composta de:

- Prova parcial 2 (**P2**) escrita, individual e sem consulta (70%) – NOTA A (TIA)
- Prova de Laboratório 2a (**Lab2a**) (20%) – NOTA B (TIA)
- Exercícios de Laboratório 2b (**Lab2b**) (10%) – NOTA C (TIA)

**Média Intermediária (MI)**

$$MI = (N1 + N2)/2 + NP$$

**Nota de participação (NP)**

NP – até um ponto (0 a 1.0) – A atividade será definida pelo professor.

**CRITÉRIOS DE APROVAÇÃO**

se  $MI \geq 7.5$  e  $FREQUÊNCIA \geq 75\%$ , **APROVADO**.

se  $MI \geq 8.5$  e  $65\% \leq FREQUÊNCIA < 75\%$ , **APROVADO**.

se  $FREQUÊNCIA \geq 75\%$  e  $(MI + PROVA FINAL)/2 \geq 6.0$ , **APROVADO**.

OBS: o aluno tem o direito de fazer uma PROVA SUBSTITUTIVA para substituir uma nota de uma avaliação que tenha se ausentado. A PROVA SUBSTITUTIVA contém todo o conteúdo do semestre. Caso o aluno tenha se ausentado em mais de uma avaliação, utilizar-se-á a nota de MAIOR PESO.

**Bibliografia Básica:**

GOODRICH, M. T.; TAMASSIA, R., MOUNT, M.N. Data Structures and Algorithms in C++. 2.ed. New York: Wiley, 2011.

SZWARCFTER, J.L.; MARKENZON, L. Estruturas de Dados e seus Algoritmos. 3ª. ed. Rio de Janeiro: LTC, 2010.

ZIVIANI, N. Projeto de Algoritmos: Com Implementações em Java e C++. São Paulo: Cengage Learning, 2011.



***Bibliografia Complementar:***

ASCENCIO, A. F. G.; ARAÚJO, G. S. Estrutura de dados: algoritmos, análise da complexidade e implementações em Java e C/C++. São Paulo: Pearson Education do Brasil, 2011.

CORMEN, T. H.; LEISERSON, C. E.; RIVEST, L.R. Introduction to algorithms. Cambridge: The MIT Press, 2000. FEOFILOFF, P. Algoritmos em linguagem C. Rio de Janeiro: Elsevier, Campus, 2009.

PUGA, S.; RISSETTI, G. Lógica de programação e estrutura de dados: com aplicações em Java. 2ª ed. São Paulo: Pearson, 2010.

VILLAS, M. V. Estruturas de dados: conceitos e técnicas de implementação. Rio de Janeiro: Campus, 2002.