



**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> FABIO APARECIDO GAMARRA. LUBACHESKI ANA GRASIELLE DIONISIO CORREA CHARLES BOULHOSA RODAMILANS MARCIO PORTO FEITOSA		<b>DRT:</b> 114633-0 1139178
<b>Carga horária:</b> 04h/a		<b>Semestre Letivo:</b> 1º sem. 2019
<b>Ementa:</b> Estudo e desenvolvimento de algoritmos envolvendo vetores, matrizes e funções, tendo com ênfase a resolução de problemas em ordem crescente de complexidade. Implementação de algoritmos utilizando linguagem de programação imperativa.		
<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>• Consolidar os conhecimentos de resolução de problemas</li><li>• Praticar com uma linguagem de programação que permita utilizar o paradigma estruturado.</li><li>• Compreender o uso das técnicas de programação em projetos.</li></ul>	<ul style="list-style-type: none"><li>• Aplicar técnicas de programação na implementação de sistemas, utilizando uma linguagem de programação.</li><li>• Identificar o potencial de sistemas computacionais na solução de problemas em diversas áreas.</li></ul>	<ul style="list-style-type: none"><li>• Ampliar a habilidade de solução de problemas.</li><li>• Reconhecer a importância do trabalho em grupo para desenvolvimento de projetos computacionais.</li></ul>



**Conteúdo Programático:**

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

0.1. Revisão de funções em Python

0.2. Revisão de listas em Python

1. Vetores

1.1. Vetores X listas em Python

1.2. Acessando, percurso e operações com vetores

2. Matrizes

1.1. Alocando, acessando e exemplos clássicos de utilização de matrizes

2.2. Modelando problemas com matrizes

3. Busca e ordenação com vetores

3.1. Busca linear e binária

3.2. Algoritmos elementares de ordenação (bolha e inserção)

4. Recursividade

4.1. Resolução de problemas recursivamente

4.2. Mergesort

4.3. QuickSort

5. PyGame

5.1. Desenhos com chamadas de funções

5.2. Eventos de teclado e mouse

5.3. Carregamento de imagens e sons

**Metodologia:**

- Aulas expositivas.
- Aulas práticas em laboratórios.
- Projeto extraclasse em equipes.
- 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)
- Laboratório 1 (**Lab1**) (30%) – NOTA B (TIA)

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

- Prova parcial 2 (**P2**) escrita, individual e sem consulta (70%) – NOTA F (TIA)
- Laboratório 2 (**Lab2**) (30%) – NOTA G (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:**

DIERBACH, C. **Introduction to Computer Science Using Python: A Computational Problem-Solving Focus**. 1.ed. New York: Wiley, 2012.

ZELLE, J.M. **Python Programming: An Introduction to Computer Science**, 2nd Edition, Franklin, Beedle & Associates Inc, 2009.

MENEZES, N.N.C. **Introdução à Programação com Python: algoritmos e lógica de programação para iniciantes**. São Paulo: Novatec, 2014.

**Bibliografia Complementar:**

PAYNE, B. **Ensine seus filhos a programar**. S.o Paulo: Novatec, 1a. ed. 2015.

LOPES, A.; GARCIA, G.; **Introdução a Programação: 500 Algoritmos**. Rio de Janeiro: Editora Campus, 2002.

FORBELLONE, A. L. V.; EBERSPACHER, H. F. **Lógica de Programação: A Construção de Algoritmos e Estrutura de Dados**. 3. ed. Sao Paulo: Pearson Prentice Hall, 2012.

PIVA Jr., D.; NAKAMITI, G.S., ENGELBRECHT, A.M. **Algoritmos e Programação de Computadores**. Rio de Janeiro: Editora Elsevier Ltda, 2012.

KINSLEY, H.; MCGUGAN, W. **Introdução ao Desenvolvimento de Jogos em Python com PyGame**. São Paulo: Novatec, 2015.