



UNIVERSIDADE PRESBITERIANA MACKENZIE
Faculdade de Computação e Informática



UNIDADE - FACULDADE DE COMPUTAÇÃO E INFORMÁTICA		
CURSO – CIÊNCIA DA COMPUTAÇÃO		
DISCIPLINA – ALGORITMOS E PROGRAMAÇÃO II		CÓDIGO DA DISCIPLINA ENEC00094
PROFESSOR(ES) Ana Grasielle Dionísio Corrêa Charles Rodamilhas	DRT 1139178	ETAPA 2ª
CARGA HORÁRIA 4 h/a (2 teoria 0 laboratório 2 EAD)		SEMESTRE LETIVO 2018/2
EMENTA 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.		
OBJETIVOS		
FATOS E CONCEITOS - Conhecer o conceito de algoritmo computacional - Identificar passos para soluções de problemas elementares e formalizá-los através de algoritmos - Avaliar e comparar soluções algorítmicas para problemas elementares - Conhecer estruturas de programação de uma linguagem imperativa e aplicá-los na implementação de algoritmos.	PROCEDIMENTOS E HABILIDADES - Construir algoritmos computacionais para problemas elementares, - Implementar algoritmos em uma linguagem de programação imperativa - - Configurar e utilizar ambientes de implementação de algoritmos - Simular implementações de algoritmos para avaliação de funcionamento e detecção de erros	ATITUDES, NORMAS E VALORES - Reconhecer a importância dos algoritmos para resolução de problemas. - Reconhecer a importância da linguagem de programação na implementação de algoritmos. - Reconhecer a área de programação como um suporte essencial na construção de sistemas computacionais. - Perceber e superar dificuldades inerentes ao pensamento algorítmico.
CONTEÚDO PROGRAMÁTICO - Funções - Matrizes - Listas - Algoritmos de busca e ordenação - Dicionários - Manipulação de arquivos		
METODOLOGIA <ul style="list-style-type: none">• Aulas expositivas.• Aulas práticas em laboratórios com de ferramentas para desenvolvimento dos projetos em Python.• Projeto em equipes.		



- Utilização do ambiente Mackenzie Virtual.

CRITÉRIO DE AVALIAÇÃO

--- N1 ---

Avaliação 1: Prova Parcial 1 (individual e sem consulta) – 70%

Avaliação 2: Projeto – 30%

--- N2 ---

Avaliação 1: Prova Parcial 1 (individual e sem consulta) – 70%

Avaliação 2: Projeto Jogo – 30%

--- Média intermediária (MI) ---

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

--- Nota de Participação (NP) ---

NP = 1,0 (máximo 0,5 – de acordo com as entregas de laboratório + máximo 0,5 – avaliação diagnóstica)

CRITÉRIOS DE APROVAÇÃO

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

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

se $FREQUENCIA \geq 75\%$ 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. New York: Wiley, 2012.

ZELLE, J.M. Python Programming: An Introduction to Computer Science. 2.ed. New York: 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. São Paulo: Novatec, 1a. ed. 2015.

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.

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

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.