



PLANO DE ENSINO

DISCIPLINA – PROGRAMAÇÃO IMPERATIVA

Código: COMP0334 N. de Créditos: 04 PEL: 2.00.2
Carga Horária: 60 Período: 2020/2 Turmas: T18

PROFESSOR - Antônio Monteiro Freire
- Mestrado em Engenharia de Sistemas e Computação (COPPE/UFRJ)

1. EMENTA

Noções fundamentais sobre algoritmos e sobre a execução de programas. Análise e síntese de problemas. Identificadores, tipos, constantes, variáveis, tipos. Operadores e expressões. Comandos condicionais e de repetição. Variáveis compostas homogêneas e heterogêneas. Procedimentos, funções e passagem de parâmetros. Noções sobre o uso de arquivos em programação. Algoritmos básicos de ordenação. Recursividade. Uma linguagem imperativa. Convenções de código. Boas práticas de programação.

2. OBJETIVOS

2.1 Geral

- A disciplina objetiva fornecer ao estudante os conhecimentos de uma linguagem de programação para especificar e escrever programas de computador.

2.2 Específicos

- Dotar o aluno do conhecimento básicos de uma linguagem de programação de propósitos gerais, que poderá ser usada como ferramenta de apoio para as demais disciplinas do curso.
- Estudar e aplicar as estruturas básicas que compõem uma linguagem de programação;

3. CONTEÚDO DAS AULAS (Por aula de 2 horas)

Data	Conteúdo
29/03/2021	01. Introdução. Plano de ensino da disciplina. Estrutura básica de um computador. (online)
31/03/2021	02. Armazenamento de dados. Representação binária. Endereço de memória.
05/04/2021	03. Programa de computador. Comando ou instrução. Linguagens de programação. (online)
07/04/2021	04. Compiladores e Tradutores. Programas e algoritmos.
12/04/2021	05. A Linguagem Python: elementos básicos, constantes, variáveis e palavras reservadas. (online)
14/04/2021	06. Elementos básicos. Constantes, variáveis e palavras reservadas.
19/04/2021	07. Expressões e operadores. Comandos de E/S. (online)
26/04/2021	08. Exercícios de programação. (online)
28/04/2021	09. Blocos de código (indentação).
03/05/2021	10. Comando condicional If. Estrutura de repetição While. (online)
05/05/2021	11. Exercícios de programação.
10/05/2021	12. Estrutura de repetição For. Comandos: break e continue. (online)
12/05/2021	13. Exercícios de programação.
17/05/2021	14. Criando e manipulando listas. (online)
19/05/2021	15. Conversão de tipos. Comentários. Exercícios de programação.
24/05/2021	16. Funções predefinidas. Bibliotecas de funções. Funções matemáticas. (online)
26/05/2021	17. Exercícios de programação.
31/05/2021	18. Compreensão de listas. (online)
02/06/2021	19. Exercícios de programação.
07/06/2021	20. Criando e manipulando tuplas. (online)
09/06/2021	21. Exercícios de programação.
14/06/2021	22. Criando e manipulando dicionários. (online)
16/06/2021	23. Exercícios de programação.
21/06/2021	24. Arrays em Python. Biblioteca NumPy. (online)
23/06/2021	25. Exercícios de programação.
28/06/2021	26. Definição de funções, retorno de valores, argumentos e parâmetros. (online)

30/06/2021	27. Exercícios de programação.
05/07/2021	28. Recursividade. (online)
07/07/2021	29. Exercícios de programação.
12/07/2021	30. Arquivos em Python (online)

Obs.: As aulas de segunda-feira serão síncronas (online).

4. METODOLOGIA

A metodologia a ser utilizada tem por objetivo manter o aluno sempre atento ao desenvolvimento da disciplina, e para tanto teremos as seguintes atividades:

- Aulas expositivas remotas síncronas e assíncronas;
- Utilização de slides durante todo o curso;
- Exercícios resolvidos durante as aulas;
- Uso de aplicativos nas aulas para edição e testes de programas;
- Exercícios de programação durante todo o curso;
- Atividades de curto prazo para casa;
- Trabalhos de programação para casa de médio prazo.

Os trabalhos e atividades feitas em casa deverão ser individuais, sendo atribuída a nota zero para programas ou atividades semelhantes. Os trabalhos não semelhantes, serão corrigidos de acordo com os seguintes critérios:

- | | |
|----------------------------|---------------------------------------|
| a) Originalidade | c) Generalidade da solução (programa) |
| b) Eficiência do algoritmo | d) Exatidão dos resultados |

5. AVALIAÇÃO (tipos e instrumentos)

A avaliação será feita através de atividades de curto prazo e de trabalhos de médio prazo para casa. As atividades de curto prazo objetivam manter os alunos acompanhando o curso de forma efetiva. Os trabalhos para casa objetivam fazer com que o aluno desenvolva seu raciocínio de forma mais ampla combinando vários temas dados em sala. Os trabalhos serão testados com o uso do computador.

1ª avaliação: atividades: 8 pontos; trabalho nº 1: 2 pontos.

2ª avaliação: atividades: 8 pontos; trabalho nº 2: 2 pontos.

3ª avaliação: atividades: 8 pontos; trabalho nº 3: 2 pontos.

6. BIBLIOGRAFIA

6.1 Básica

- Como pensar como um Cientista da Computação usando Python (traduzido). Allen Downey, Jeffrey Elkner, and Chris Meyers.
- Python para Desenvolvedores. Luiz Eduardo Borges. Rio de Janeiro; 2010; 978-85-909451-1-6.
- Introdução à Programação com Python. Nilo Ney Coutinho, 2ª edição, 2014, ISBN: 978-85-7522-408-3.

6.2 Complementar

- Fundamentos da Programação de Computadores. Ana Fernanda Gomes Ascencio / Edilene Aparecida Veneruchi De Campos. 3ª edição; 2012, Pearson; ISBN 978-8564574168

7. OBSERVAÇÕES

Critérios de correção das atividades e trabalhos feitos em casa:

Os exercícios feitos em casa deverão ser testados no computador, sendo atribuída a nota zero para programas muito semelhantes. Os exercícios serão corrigidos de acordo com os seguintes critérios:

- Originalidade
- Eficiência do algoritmo
- Generalidade do programa
- Exatidão dos resultados

Campus, 22 de março de 2021.

Antônio Monteiro Freire
Professor