



UNIVERSIDADE FEDERAL DE UBERLÂNDIA



## FICHA DE COMPONENTE CURRICULAR

<b>CÓDIGO:</b>	<b>COMPONENTE CURRICULAR:</b> ESTRUTURAS DE DADOS I	
<b>UNIDADE ACADÊMICA OFERTANTE:</b> FACULDADE DE COMPUTAÇÃO		<b>SIGLA:</b> FACOM
<b>CH TOTAL TEÓRICA:</b> 30 horas	<b>CH TOTAL PRÁTICA:</b> 30 horas	<b>CH TOTAL:</b> 60 horas

1. **OBJETIVOS**

Capacitar o aluno a escrever programas eficientes, em termos de tempo e espaço, utilizando boas práticas de programação, para a solução de problemas baseados em listas, filas e pilhas. Conhecer os principais algoritmos de ordenação e busca binária.

2. **EMENTA**

Pilhas, filas, listas encadeadas. Introdução à análise de algoritmos. Busca Binária. Ordenação.

3. **PROGRAMA****1. Definição de Classes utilizando a linguagem Python**

1.1. Conceitos de programação orientada a objetos.

1.2. Implementação de classe para representar TAD (Tipo Abstrato de Dados).

**2. Estruturas de dados lineares**

2.1. Pilha.

2.1.1. Definição baseada em *array*. Conceitos, restrições de acesso e operações.

2.1.2. Implementação do TAD Pilha em Linguagem Python.

2.1.3. Aplicações.

2.2. Fila.

2.2.1. Definição baseada em *array*. Conceitos, restrições de acesso e operações.

2.2.2. Implementação do TAD Fila em Linguagem Python.

2.2.3. Aplicações.

2.3. Listas.

2.3.1. Possibilidades de TADs para Listas.

2.3.2. Lista Sequencial.

2.3.2.1. Definição baseada em *array*. Conceitos, operações considerando lista ordenada e lista não ordenada.

- 2.3.2.2. Implementação do TAD Lista Sequencial sem ordenação em Linguagem Python.
- 2.3.2.3. Implementação do TAD Lista Sequencial com ordenação em Linguagem Python.
- 2.3.3. Lista Encadeada Simples.
  - 2.3.3.1. Conceitos, operações para lista ordenada e não ordenada.
  - 2.3.3.2. Implementação do TAD Lista Encadeada Simples sem ordenação em Python.
  - 2.3.3.3. Implementação do TAD Lista Encadeada Simples com ordenação em Python.
  - 2.3.3.4. Aplicações
- 2.3.4. Lista Encadeada Circular.
  - 2.3.4.1. Conceitos e Implementação do TAD Lista encadeada Circular em Linguagem Python.
- 2.3.5. Lista Duplamente Encadeada.
  - 2.3.5.1. Conceitos e Implementação do TAD Lista Duplamente Encadeada em Python.

### **3. Introdução à análise de algoritmos.**

#### **4. Busca e Ordenação em Listas**

- 4.1. Introdução à Análise de Algoritmos.
  - 4.1.1. Analisar o tempo de execução de uma função em relação ao tamanho da entrada de dados.
  - 4.1.2. Funções clássicas de comportamento de um algoritmo.
- 4.2. Busca Binária em Lista Sequencial.
- 4.3. Métodos de ordenação.
  - 4.3.1. Bolha.
  - 4.3.2. Seleção.
  - 4.3.3. Inserção.
- 4.4. Métodos com estratégia 'dividir para conquistar'.
  - 4.4.1. *Quicksort*.
  - 4.4.2. *MergeSort*.
- 4.5. Comparação entre os métodos de ordenação.

#### **4. BIBLIOGRAFIA BÁSICA**

1. GOODRICH, M.T.; TAMASSIA, R.; GOLDWASSER, M.H. **Data structures and algorithms in Python**. John Wiley & Sons Ltd, 2013.
2. BAKA, B. **Python Data Structures and Algorithms: Improve application performance with graphs, stacks, and queues**. Packt Publishing, 2017.
3. LUTZ M. **Learning Python**. O'Reilly, 2013.

#### **5. BIBLIOGRAFIA COMPLEMENTAR**

1. DROZDEK, A. **Estrutura de dados e algoritmos em C++**. São Paulo: Cengage Learning, c2002.
2. MENEZES N. C. **Introdução à Programação com Python**. Novatec 2010.
3. MEHLHORN, K. **Data structures and algorithms**. Berlin: Springer, 1984.
4. SEDGEWICK, R. **Algorithms in C**. 3rd. ed. Boston: Addison-Wesley, c1998-2002.
5. TENENBAUM, A. M. **Estruturas de dados usando C**. São Paulo: Makron Books, 1995.

#### **6. APROVAÇÃO**

Prof. Dr. Jean Carlos Domingos  
Coordenador do Curso de Graduação em Gestão  
da Informação

Prof. Dr. Mauricio Cunha Escarpinati  
Diretor da Faculdade de Computação



Documento assinado eletronicamente por **Jean Carlos Domingos, Coordenador(a)**, em 18/11/2019, às 15:52, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por **Mauricio Cunha Escarpinati, Diretor(a)**, em 18/11/2019, às 17:06, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site [https://www.sei.ufu.br/sei/controlador\\_externo.php?acao=documento\\_conferir&id\\_orgao\\_acesso\\_externo=0](https://www.sei.ufu.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0), informando o código verificador **1660353** e o código CRC **DE6D8F6F**.

Referência: Processo nº 23117.094672/2019-84

SEI nº 1660353