



UNIVERSIDADE FEDERAL DE UBERLÂNDIA

FICHA DE COMPONENTE CURRICULAR

CÓDIGO:	COMPONENTE CURRICULAR: PROGRAMAÇÃO SCRIPT	
UNIDADE ACADÊMICA OFERTANTE: FACULDADE DE ENGENHARIA ELÉTRICA	SIGLA: FEELT	
CH TOTAL TEÓRICA: 30 horas	CH TOTAL PRÁTICA: 30 horas	CH TOTAL: 60 horas

OBJETIVOS

Objetivo Geral: Desenvolver as habilidades dos discentes em programação de computadores para resolução de problemas em suas rotinas pessoais e profissionais através do emprego de linguagens de script.

Objetivos Específicos:

1. Desenvolver a compreensão de conceitos fundamentais de programação e de computação;
2. Desenvolver as habilidades dos discentes tanto em trabalhar em grupo, gerenciar projetos e documentá-los quanto em produzir códigos de qualidade através da adoção de boas práticas de programação e de *Test-Driven Development* (TDD);
3. Desenvolver a consciência, a autonomia e a responsabilidade dos discentes;
4. Tornar prazerosa para os discentes a atividade de programar computadores.

EMENTA

Desenvolvimento de aplicações práticas em linguagens de script, com abordagem algorítmica e conceitos de avaliação de valores e de expressões; constantes, variáveis, *strings*, *arrays* e objetos; controle de fluxo de execução de código (sequência, instruções condicionais, laços, funções, encapsulamento). Noções de estrutura de dados, interpretadores, análise de algoritmos, computabilidade e complexidade. Conceitos de computação nas nuvens. Noções de empreendedorismo e de inovação.

PROGRAMA

1. **Introdução**
 - 1.1. Programação de computadores
 - 1.2. Linguagens de Script e Interpretadores
 - 1.3. Repositórios (Github) e plataformas de desenvolvimento (servidores e computação nas nuvens)
 - 1.4. Aplicativos e inovação
2. **Programação Web**
 - 2.1. HTML e CSS
 - 2.2. Primitivas, tipos de dado e estruturas: constantes, variáveis, *strings*, *arrays*
 - 2.3. Instruções condicionais, laços, funções
 - 2.4. Estrutura de dados composta: objeto JSON

- 2.5. Implementação e análise de algoritmos, estudo de complexidade
 2.6. *Rapid application development* (RAD): Node-RED, Django
 2.7. Desenvolvimento de aplicações
- 3. Linguagem Python**
- 3.1. Primitivas, tipos de dado e estruturas: variáveis, *strings*, *arrays*
 - 3.2. Instruções condicionais, laços, funções, recursão
 - 3.3. Estruturas de dados compostas: Listas, arquivos, dicionários, classes
 - 3.4. Implementação e análise de algoritmos, estudo de complexidade
 - 3.5. Tópicos avançados, desenvolvimento de aplicações em nuvem
- 4. Cursos online abertos e massivos**
- 4.1. HTML & CSS: <https://www.codecademy.com/learn/learn-html-css>
 - 4.2. Javascript:
 - 4.2.1. <https://br.udacity.com/course/javascript-basics--ud804/>
 - 4.2.2. <https://www.codecademy.com/learn/learn-javascript>
 - 4.3. Python:
 - 4.3.1. <https://br.udacity.com/course/programming-foundations-with-python--ud036/>
 - 4.3.2. <https://www.codecademy.com/pt-BR/learn/python>

BIBLIOGRAFIA BÁSICA

LUTZ, Mark. **Aprendendo Python**. Porto Alegre: Bookman, 2007.

FREEMAN, Elisabeth. **Use a cabeça!**: HTML com CSS e XHTML. 2. ed. Rio de Janeiro: Alta Books, 2008.

MENEZES, Nilo Ney Coutinho. **Introdução à programação com Python**: algoritmos e lógica de programação para iniciantes. São Paulo: Novatec, 2010.

BIBLIOGRAFIA COMPLEMENTAR

DEWDNEY, A. K. **The (new) Turing Omnibus**: 66 excursions in computer science. New York: W.H. Freeman, 2001.

LINDLEY, Cody. **JavaScript Succinctly**. Syncfusion, Inc., 2017. Versão online disponível em: <<https://www.syncfusion.com/ebooks/javascript>>. Acesso em: 28 jun. 2018.

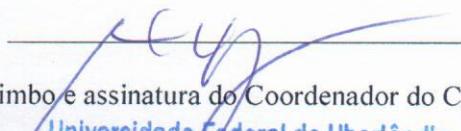
LUTZ, Mark. **Aprendendo Python**. Porto Alegre: Bookman, 2007.

MILLER, Brad; RANUM, David. **Como pensar como cientista da computação**: aprendendo com Python: edição interativa (usando Python 3.x). 2017. Disponível em: <<https://panda.ime.usp.br/pensepy/static/pensepy/>>. Acesso em: 28 jun. 2018.

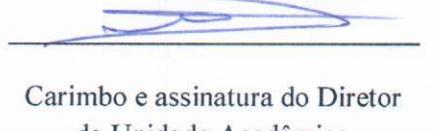
SWEIGART, Al. **Automate the boring stuff with Python**. 2015. Versão online. Disponível em: <<https://automatetheboringstuff.com/>>. Acesso em: 28 jun. 2018.

APROVAÇÃO

06/09/18


 Carimbo e assinatura do Coordenador do Curso
Universidade Federal de Uberlândia
 Prof. Dr. Marcelo Rodrigues de Souza
 Coordenador do Curso de Engenharia da Computação
 Portaria R. N° 1234/2017

10/09/18


 Carimbo e assinatura do Diretor
 da Unidade Acadêmica
Universidade Federal de Uberlândia
 Prof. Dr. Sérgio Ferreira de Paula Silva
 Diretor da Faculdade de Engenharia Elétrica
 Portaria R. N° 708/17