

## LOM3260 - Computação Científica em Python

### *Scientific Computing in Python*

- Créditos-aula: 4
- Créditos-trabalho: 0
- Carga horária: 60 h
- Departamento: Engenharia de Materiais

### Objetivos

Fornecer ao aluno uma introdução à computação científica moderna, usando a linguagem Python e suas bibliotecas numéricas e gráficas mais populares: numpy, scipy, matplotlib e pandas. Ao final do curso, o aluno estará capacitado a desenvolver programas complexos, de pequeno e médio porte para solucionar problemas de engenharia que envolvam processamento numérico de grandes conjuntos de dados e correlacionar variáveis usando métodos numéricos.

### Docente(s) Responsável(eis)

- 519033 - Carlos Yujiro Shigue
- 1176388 - Luiz Tadeu Fernandes Eleno

### Programa resumido

Introdução à programação em Python; palavras-chave em Python; rotinas e funções; classes; numpy e o conceito de slicing e indexing de arrays; revisão de métodos numéricos usando scipy; geração de gráficos e animações com a biblioteca matplotlib; criação de interfaces gráficas com o usuário usando matplotlib.widgets

### Programa

• Introdução à programação em Python o Instalação de uma distribuição Python em Windows e Linux o Formatação de arquivos em Python o Estruturas condicionais o Laços de repetição de comandos o Outras palavras-chaves e métodos o Rotinas e funções o Códigos multifonte e bibliotecas pessoais o Bibliotecas numéricas e gráficas: numpy, scipy e matplotlib • Programação orientada a objeto: classes o Conceito de objetos e instâncias o Classes e subclasses • “Arrays” em numpy o O conceito de array em numpy o “Slicing” e indexação o Trabalhando com arquivos (entrada e saída) • Métodos numéricos em scipy o Zero de funções o Resolução numérica de integrais o Ajuste não-linear de funções a um conjunto de dados • Gráficos em matplotlib o A biblioteca matplotlib.pyplot e gráficos em 2D e 3D o A biblioteca matplotlib.animation para criar gráficos animados. • Interfaces gráficas com o usuário (Graphical User Interface, GUI) o Interfaces simples com a biblioteca matplotlib.widgets.

### Avaliação

- **Método:** Aulas expositivas e em laboratório computacional, trabalhos e exercícios comentados.
- **Critério:** Média aritmética de trabalhos propostos ao longo do curso.
- **Norma de recuperação:** Não haverá exame de recuperação.

### Bibliografia

• Nilo Ney Coutinho Menezes. Introdução à Programação com Python: Algoritmos e Lógica de Programação Para Iniciantes, 3a ed, 2019.  
• LANGTANGEN, Hans Petter. A Primer on scientific programming with Python, 2a ed. New York: Springer, 2011. • LANGTANGEN, Hans Petter. Python scripting for computational science, 5a ed. New York: Springer, 2016. • STEWART, J. M. Python for scientists. Cambridge University Press, 2014.  
• TELLES, M. Python Power, Boston: Thomson Course Technology PTR, 2008. • LUTZ, Mark. Programming Python, 3a ed, Sebastopol, CA: O’Reilly Media, 2006. • SCOPATZ, A.; HUFF, K. D. Effective computation in physics: field guide to research in Python. Sebastopol, CA: O’Reilly Media, 2015.  
• KINDER, J. M. A Student’s guide to Python for physical modeling. Princeton University Press, 2015. • MCGREGGOR, D. M. Mastering matplotlib. Birmingham, UK: Packt Publishing, 2015. NumPy community, Numpy Reference, 2014. • Scipy community, Scipy Reference Guide, 2017. • HUNTER, J.; DALE, D.; FIRING, E.; DROETTBOOM, F. M. Matplotlib manual, 2016.

[Ver no Jupiter](#) [Salvar em pdf](#) [Salvar em docx](#)