

Notas de aula

# Introdução à Física Computacional

Prof. Gerson – UFU – 2019

Atendimento:

- Sala 1A225
- Email: [gersonjferreira@ufu.br](mailto:gersonjferreira@ufu.br)
- Webpage: <http://gjferreira.wordpress.com>
- Horário: \_\_\_\_\_

# Antes de começarmos...

---

Acesse o site da distribuição Anaconda e baixe o instalador.

→ <https://www.anaconda.com>

- Windows (64 bits)
- Linux: 64-Bit (x86) Installer (517 MB)

No Linux, após instalação copiar arquivo  
spyder3.desktop para /home/aluno/.local/share/applications  
o arquivo spyder3.desktop esta disponível na minha página.

# Proposta do curso

Resolver problemas clássicos da física usando técnicas numéricas

## **Problemas:**

- Equações diferenciais (massa-mola, planetas, dinâmica molecular, Schrödinger, difusão)
- Análise espectral / espectroscopia (transformadas de Fourier)
- Estatística (Monte Carlo, Ising, Machine Learning)
- Interação com hardware (microfone, camera, ...)

## **Técnicas:**

- Euler, Runge-Kutta, Verlet → cálculo numérico
- Fast Fourier Transform (FFT)
- Álgebra Linear numérica (autovalores/autovetores, sistemas lineares)
- Integração/quadratura numérica
- Detecção de imagens, análise e manipulação de sinais

# Linguagem de programação



ANACONDA®

NUMFOCUS  
OPEN CODE = BETTER SCIENCE

matplotlib

## Por que python?

- Fácil de aprender
- Acessível
- Infinitude de pacotes complementares úteis
- É o mais usado fora da academia = emprego

## Além de Python?

- Aprender C ou Fortran
- LabView / Matlab
- Mathematica / Maple
- Computação de alto desempenho

# Bibliografia

Tutoriais, textos, vídeo-aulas... estarão disponíveis na minha página

- Livros de introdução a programação em python
  - há vários livros na biblioteca, escolham alguns e tragam na próxima aula porém todos devem cobrir os tópicos introdutórios:
    - Estrutura da linguagem
    - If/else
    - Loops for/while
    - Funções
    - Tipo e escopo de variáveis
    - ...
- Livros de cálculo numérico: veja o meu livro na minha página.

# Critérios de avaliação

- Presença: 75%
- Listas de exercício: introdução a cada tópico [10 pts]
- Projetos em python [70 pts]
- Projeto final [20 pts], poderão escolher:
  - Projeto “simples”, mas feito em outra linguagem: C / Fortran / Mathematica / Maple
  - Projeto de computação concorrente/paralela (em Python ou C)
  - Uso de ferramentas avançadas:
    - Quantum Espresso (DFT)
    - LAMMPS (dinâmica molecular)
    - Machine Learning
  - Outras sugestões?
- Aprovação: nota final > 60

# Aula de hoje

- 1) Breve conversa sobre o curso, expectativa e interesse de vocês
- 2) Baixar, instalar e testar o Anaconda Python  
sugestão: usaremos o editor **Spyder IDE** ← já vem como Anaconda

Testar se tudo está funcionando com o código:

```
import numpy as np
import matplotlib.pyplot as plt

x = np.linspace(0, 2*np.pi, 101)
y = np.sin(x)

plt.plot(x, y)
plt.show()
```