

INTRODUCCIÓN A MÉTODOS NUMÉRICOS PARA FÍSICA TEÓRICA

Semestre: enero-junio 2025

Impartido por: Armando A. Roque Estrada.

Para inscribirse en el curso, enviar un correo con sus datos a la dirección al calce del documento. Como requisitos se ha de estar cursando mínimo el 5to semestre de licenciatura.

Descripción del curso:

Las clases combinarán indistintamente conferencias y clases prácticas (implementación de códigos numéricos). A continuación, se presentan algunos detalles del curso:

Se abordarán las bases conceptuales (no se profundizará en demasía) de los aspectos teóricos que se implementarán en el cursos . Se cubrirá un conjunto de tópicos que permitirá profundizar en conocimientos de programación y matemáticas, además de servir como una herramienta fundamental para futuros trabajos científicos.

Como lenguaje principal, usaremos Python; sin embargo, en casos puntuales, podremos utilizar otros lenguajes. El curso se dividirá en dos bloques principales y un bloque de introducción a Python. Toda la información se puede encontrar en el repositorio: Github

Bloque Cero (El ABC de Python)

- Buenas prácticas al programar.
- Ideas básicas sobre Python y algunos paquetes fundamentales.

Primer Bloque:

- Números.
- Derivadas.
- Matrices
- Roots.

Segundo Bloque:

- Interpolación.
- Fitting.
- Integrales.
- Ecuaciones diferenciales ordinarias en el contexto de equilibrio hidrostático.
- Ecuaciones diferenciales parciales. (tentativo)
- Redes neuronales. (tentativo)

Para cualquier duda pueden contactarme por mi correo personal:

arestrada@fisica.uaz.edu.mx

Fuentes de Información

1. Numerical Methods in Physics with Python, Alex Gezerlis. (2020, Cambridge University Press)
2. Python Programming and Numerical Methods A Guide for Engineers and Scientist, Qingkai Kong, Timmy Siau, Alexandre Bayen.

Acondicionamiento del equipo de cómputo

En caso de querer instalar **Python 3.x** a continuación se brindan unos consejos para su instalación:

Los consejos que se comentan a continuación sirven de guía para instalar en Linux, Mac OS o Windows mediante el software (gestor de paquetes) Anaconda. Una alternativa puede ser descargar directamente Python: <https://www.python.org> y el gestor de paquetes PIP (ver tutorial <https://acortar.link/RG5PS8>).

Pasos:

- 1- Descargar e instalar la versión de Anaconda correspondiente al sistema operativo y versión de Python usada (se recomienda python 3.)

<https://www.anaconda.com/products/individual>

- 3- Configurar Anaconda,

(1) Actualizar *conda* y *jupyter*, abrir una terminal y teclear,

conda update conda

conda update jupyter

- (2) Instalar librerías básicas,

conda install anaconda::numpy (<https://numpy.org>)

conda install conda-forge::matplotlib (<https://matplotlib.org>)

conda install anaconda::scipy (<https://scipy.org>)

conda install anaconda::pandas (<https://pandas.pydata.org>)

Otras: TensorFlow, PyTorch, Keras, Scikit-learn, Seaborn, Bokeh, SymPy, Numba

- (3) Comprobar la instalación, escribir en una terminal y teclear *jupyter-*

notebook

Posibles errores:

No se ejecuta, python, conda o jupyter-notebook desde la consola. Posible solución:

Windows

<https://acortar.link/RHLKrH>

MacOS y Linux

Es necesario modificar el `.bash_profile`, contactarme por correo para explicarles como hacerlo.

En caso de no querer instalar Python en su equipo de cómputo, pueden hacer uso de la plataforma

Colaboratory: <https://colab.research.google.com/?hl=es>

la cual cuenta con la mayoría de los recursos necesarios para un buen desarrollo del curso. En necesario señalar que han de tener una cuenta en gmail para su uso.

EDITOR

Una de las herramientas más útiles para programar son los editores, los mismos nos ahorran mucho trabajo. Aunque para el caso de Python podemos usar (y usaremos) la herramienta jupyter, pero cuando se trabaja con servidores, etc. es necesario crear *scripts* con nuestros códigos, siendo los editores la herramienta ideal para ello. Uno que recomiendo es Visual Studio Code <https://code.visualstudio.com> con las extensiones: Python, Jupyter, Python Indent, Python Extended.