# 4A Problemario: Solución Numérica de Sistemas de Ecuaciones Diferenciales Ordinarias

* Maestría en Sistemas Inteligentes Embebidos
* Materia: Métodos matemáticos
* Unidad: Métodos numéricos
* Docente: Dr. Juliho Castillo Colmenares
* Puntaje total: 15

## Objetivo

Aplicar y analizar diferentes métodos numéricos para resolver sistemas de ecuaciones diferenciales ordinarias.

## Instrucciones

1. **Consulta**[Numerical Methods for Engineers.](https://www.math.hkust.edu.hk/~machas/numerical-methods-for-engineers.pdf).
2. **Recopila** y **analiza** tu información.
3. **Descarga** el archivo [4A Métodos numéricos](https://github.com/julihocc/msie-metodos-matematicos-actividades/tree/main/u4-actividad)
4. **Documenta** cada paso de tu proceso de resolución, incluyendo las ecuaciones utilizadas, los cálculos realizados y las soluciones obtenidas
5. **Utiliza** Python o SageMath para realizar los cálculos necesarios.
6. **Utiliza** Scipy para verificar que tus respuestas con correctas.
7. **Redacta** tu trabajo en una libreta Jupyter.
8. **Exporta y entrega** tu trabajo en formato PDF.
9. **Considera** los [criterios de evaluación](https://github.com/julihocc/msie-metodos-matematicos-actividades/tree/main/u4-actividad)

### 1. Solución de Ecuaciones No Lineales (Bisección y Newton-Raphson)

Considera la función f(x)=x3−6x2+11x−6f(x) = x^3 - 6x^2 + 11x - 6f(x)=x3−6x2+11x−6. Usa dos métodos diferentes para encontrar una raíz en el intervalo [2.5,4][2.5, 4][2.5,4]: el método de la bisección y el método de Newton-Raphson.

1. Implementa el **método de bisección** en SageMath o Python para encontrar la raíz de f(x)f(x)f(x) en el intervalo [2.5,4][2.5, 4][2.5,4] con una precisión de 10−610^{-6}10−6.
2. Luego, utiliza el **método de Newton-Raphson** para encontrar la misma raíz, empezando desde el punto inicial x0=3.5x\_0 = 3.5x0​=3.5.
3. Compara los resultados obtenidos con ambos métodos y discute la velocidad de convergencia de cada uno.

### 2. Métodos de Integración Numérica (Trapecio y Simpson)

Considera la función f(x)=e−x2f(x) = e^{-x^2}f(x)=e−x2 en el intervalo [0,1][0, 1][0,1].

1. Implementa el **método del trapecio** para aproximar la integral de f(x)f(x)f(x) en el intervalo [0,1][0, 1][0,1] con n=100n = 100n=100 subintervalos.
2. Repite el ejercicio anterior utilizando el **método de Simpson**.
3. Compara los resultados obtenidos con ambos métodos y discute la precisión de cada uno.

### 3. Solución de Ecuaciones Diferenciales Ordinarias (Euler y RK4)

Considera la ecuación diferencial ordinaria:

\frac{dy}{dx} = y - x^2 + 1

con la condición inicial y(0)=0.5y(0) = 0.5y(0)=0.5.

1. Usa el **método de Euler** para aproximar la solución en el intervalo [0,2][0, 2][0,2] con un paso h=0.2h = 0.2h=0.2.
2. Implementa el **método de Runge-Kutta de cuarto orden (RK4)** para resolver la misma ODE.
3. Grafica ambas aproximaciones junto con la solución exacta de la ecuación diferencial. Puedes utilizar SageMath para encontrar la solución exacta.

## Bibliografía

1. Chasnov, Jeffrey R. (2020). [Numerical Methods for Engineers.](https://www.math.hkust.edu.hk/~machas/numerical-methods-for-engineers.pdf)
2. Fuhrer, C., Solem, J. E., Verdier, O. (2021). Scientific Computing with Python - Second Edition: High-Performance Scientific Computing with NumPy, SciPy, and Pandas. India: Packt Publishing.
3. Johansson, R. (2018). Numerical Python: Scientific Computing and Data Science Applications with Numpy, SciPy and Matplotlib. Germany: Apress.
4. Linge, S., Langtangen, H. P. (2016). Programming for Computations - Python: A Gentle Introduction to Numerical Simulations with Python. Germany: Springer International Publishing.
5. Tveito, A., Langtangen, H. P., Nielsen, B. F., Cai, X. (2010). Elements of Scientific Computing. Germany: Springer.