

## Tarea 2: Cuadratura Gaussiana, documentación de código y Git (6%)

Marlon Brenes\*

### I. CUADRATURA GAUSSIANA (3.0%)

El propósito de esta sección es aplicar el concepto de cuadratura Gaussiana para integración utilizando los polinomios de Legendre.

- Escriba un script de python `quadrature.py` que resuelve la siguiente integral utilizando el método de cuadratura Gaussiana vista en clase (Tema\_04/GaussianQuadratureAndFitting.ipynb):

$$I = \int_1^3 dx [x^6 - x^2 \sin(2x)] . \quad (1)$$

Puede utilizar las mismas rutinas que se utilizan en el documento Tema\_04/GaussianQuadratureAndFitting.ipynb. ¿Con cual valor de  $N$  se alcanza el resultado exacto?

- Script de python que ejecuta el método numérico y devuelve el resultado correcto: 1.5%
- Función que devuelve los pesos y puntos de colocación: 0.5%
- Función que escala el intervalo de integración: 0.5%
- Pruebas con distintos valores de  $N$  hasta alcanzar el resultado correcto: 0.5%

### II. DOCUMENTACIÓN Y GIT (3.0%)

El propósito de esta tarea es practicar los conceptos de control de versiones bajo la implementación de `Git` y la creación de documentos de referencia utilizando `MkDocs`. Su resultado final es un link (URL) que incluye la documentación, utilizando su cuenta de GitHub como host para la página web. Usted debe entregar:

- Un URL con el cual se accesa de manera pública su página web que contiene la documentación del código (3.0%)

Los pasos para generar su página web que contiene la documentación son los mismos que seguimos en clase.

Si necesita una referencia para el uso de `MkDocs`, utilice este tutorial para más información en como generar documentación con `MkDocs`: <https://realpython.com/python-project-documentation-with-mkdocs/> (ignore lo referente a `Type Hints`). Es posible que tenga que crear una cuenta para visualizar el artículo.

#### Descripción

El objetivo es documentar la implementación del método de cuadratura Gaussiana implementado en el apartado anterior. Su documentación debe incluir los siguientes apartados:

1. `index.md`: Introducción al problema a resolver (recuerde que estos documentos usan el mismo lenguaje `Markdown`). (0.5%)
2. `explanation.md`: Describe el método numérico a utilizar, con la información que usamos en clase (0.75%)

---

\* [marlon.brenes@ucr.ac.cr](mailto:marlon.brenes@ucr.ac.cr)

3. `tutorials.md`: Incluye un ejemplo de uso (0.75%)
4. `reference.md`: contiene la documentación de las funciones. Debe ser generado de forma automática con los `docstrings` que contiene el módulo como tal. Los `docstrings` deben contener ejemplos de uso. Usted debe escribir las funciones y documentar su uso, argumentos esperados y resultados de salida. (1.0%)

Utilice el estilo de `docstrings` (PEP 257-compliant) que utilizamos en clase (ver `sum.py` en Tema\_06 del repositorio del curso).

**Usted solo debe entregar el URL de su página web con el host en GitHub, que contiene toda la información del módulo.**

### Apéndice

MkDocs se puede integrar con lenguaje LaTeX de forma muy sencilla. Sin embargo; para esto, debe añadir las siguientes cláusulas al archivo de configuración `mkdocs.yml`:

```
markdown_extensions:
  - pymdownx.arithmatex:
      generic: true

extra_javascript:
  - javascripts/mathjax.js
  - https://unpkg.com/mathjax@3/es5/tex-mml-autoload.js
```