

La solución a este taller debe subirse por SICUA antes de las 10:00PM del sábado 16 de septiembre de 2017 (**No** se reciban tareas enviadas tarde ni por otros medios distintos a SICUA). Los archivos código fuente deben subirse en un único archivo `.zip` con el nombre `NombreApellido_hw1.zip`, por ejemplo yo debería subir el zip `VeronicaArias_hw1.zip`, este archive debe descomprimirse en un directorio de nombre `NombreApellido_hw1` y sólo debe contener los códigos fuente: `MasaSol.py`, `PlotsTolima.py`, `Tolima.sh` e `integral.py` (10 puntos). Recuerden que es un trabajo totalmente individual y que deben usar siempre sus propias implementaciones numéricas.

1. (30 points) Estimación de la masa del Sol

En los archivos `MarsOrbit.dat` y `EarthOrbit.dat` se encuentran respectivamente las posiciones en x, y, z de la Tierra y Marte (columnas 2, 3 y 4) en función del tiempo (columna 1). El sistema de referencia está centrado en el Sol, el tiempo está expresado en años, y las distancias en AU (unidades astronómicas, donde 1AU corresponde aproximadamente a la distancia tierra-sol: $1\text{AU}=149597870700\text{m}$).

En este ejercicio deben escribir una rutina en Python llamada `MasaSol.py` que les permita calcular la masa del Sol a partir de las orbitas de los dos planetas. Para esto deben usar la información de las posiciones de los planetas y la relación para la fuerza gravitacional:

$$\vec{F}_{12} = G \frac{m_1 m_2}{\|\vec{r}_{12}\|^3} (\vec{r}_2 - \vec{r}_1) \quad (1)$$

La rutina de Python `MasaSol.py` debe:

- Leer los archivos de datos (`MarsOrbit.dat` y `EarthOrbit.dat`) y guardar las variables relevantes en arrays.
- Graficar en una misma gráfica las órbitas de los dos planetas (la de la tierra en color negro y la de marte en color azul). Esta gráfica debe ser clara, con *labels* para las distintas curvas y ejes debidamente rotulados.
- Guardar la gráfica anterior (sin mostrarla) en `Orbitas.pdf`.
- A partir de los datos de posiciones, calcular la masa del sol usando derivación numérica (usando una implementación escrita por ustedes).
- El código debe imprimir: "La masa del Sol obtenida a partir de las posiciones de la Tierra es XX kg y la obtenida a partir de las posiciones de Marte es YY kg", donde XX y YY son los valores encontrados por su código. Expresa el valor de la masa en kg y sea cuidadoso con el manejo de unidades.

2. (20 points) Analisis de datos del nevado del Tolima

En `DatosTolima.dat` se encuentran los datos de actividad volcánica del Nevado del Tolima entre los años 2006 y 2012. Escriba un script de bash (llamado `Tolima.sh`) y un script de Python `PlotsTolima.py` que grafique datos. El script `Tolima.sh` debe:

- Crear un directorio llamado `Tolima` y entrar a dicho directorio

- Copiar los archivos `DatosTolima.dat` y `PlotsTolima.py` al directorio Tolima
- A partir de `DatosTolima.dat` generar un archivo llamado `DatosMarzo.txt` que contenga los datos de *Glacier&&RockFall* y de *Largest Earthquake* de los meses de marzo.
- A partir de `DatosTolima.dat` generar un archivo llamado `GRF_vs_EQ.txt` que contenga los datos de *Glacier&&RockFall* y de *Largest Earthquake* de todos los meses. Asegurese de que está guardando las columnas de datos que le interesan para TODAS las líneas.
- Correr la rutina `PlotsTolima.py` en python
- Borrar el archivo `DatosTolima.dat`

La rutina de Python `PlotsTolima.py` debe:

- Leer los archivos `DatosMarzo.txt` y `GRF_vs_EQ.txt` y guardar las variables relevantes en arrays
- Graficar en una misma gráfica *Glacier&&RockFall* en función de *Largest Earthquake* para todos los meses (en color negro) y para los meses de marzo (en color verde y con puntos 10 veces más grandes que en el caso anterior). Esta gráfica debe ser clara, con *labels* para los distintos datos y ejes debidamente rotulados.
- Guardar la gráfica anterior (sin mostrarla) en `PlotTolima.pdf`

3. (20 points) **Integración numérica** El objetivo de este ejercicio es calcular numéricamente una integral. La función que deben integrar es:

$$f(x_1, x_2) = (x_1 + \cos(x_2) * x_1)^3 \quad (2)$$

Escriba un script llamado `integral.py` que:

- Calcule la integral:

$$I = \int_0^1 \int_0^\pi f(x_1, x_2) dx_1 dx_2 \quad (3)$$

de la función con al menos dos métodos de integración. El código debe imprimir la frase *el valor de la integral con el método X es XX y con el método Y es YY*.