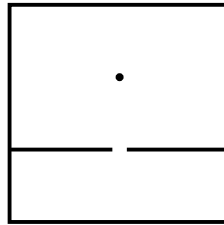


La solución a este taller debe subirse por SICUA antes de las 10:00PM del jueves 20 de julio del 2017. Los archivos código fuente deben subirse en un único archivo `.zip` con el nombre `NombreApellido_hw3.zip`, por ejemplo yo debería subir el zip `VeronicaArias_hw3.zip`. Este archivo debe descomprimirse en un directorio de nombre `NombreApellido_hw3` que sólo contenga los códigos fuente, adicionalmente éstos deben estar en un repositorio de Github (con varios commits que muestren la evolución del trabajo) cuyo enlace también deben subir a Sicua (5 puntos). Recuerden que es un trabajo totalmente individual.

1. (30 points) Ecuación de onda en 2 dimensiones

En este ejercicio deben solucionar la ecuación de onda en dos dimensiones para estudiar la propagación de una onda en un líquido contenido en contenedor cuadrado con una barrera y una rendija. Para este ejercicio deben escribir un código en python.



El código `Onda.py` debe:

- Solucionar la ecuación de onda en 2 dimensiones:

$$\frac{1}{c^2} \frac{\partial^2 \Phi(t, x, y)}{\partial t^2} = \frac{\partial^2 \Phi(t, x, y)}{\partial x^2} + \frac{\partial^2 \Phi(t, x, y)}{\partial y^2} \quad (1)$$

para una onda que se propaga en un contenedor cuadrado de con una rendija (a 1/3 de uno de los lados) como la que se muestra en la figura. Las condiciones iniciales son tales que en un punto (a 1/3 de otro de los lados como se puede ver en la figura) hay una perturbación de la superficie del líquido. Suponga que en unidades propias del problema, el contenedor tiene lados de 30, una profundidad de 2 y la perturbación una profundidad de -0.5. Suponga además que $c=1$. Use una grilla de 300 puntos, calcule el Δt a partir de la condición de estabilidad, de Δx y Δy , y corra su simulación hasta $t_{final}=60$.

- El código debe también hacer graficas de $\Phi(t, x, y)$ para $t=30$ y $t=60$. Estas gráficas deben luego ser incluidas en `Resultados_hw3.pdf`.
- Finalmente, el código debe también hacer una animación de la propagación de la onda y guardarla en `Onda.mp4`.

2. (35 points) Sistema Solar

Deben escribir un programa en C llamado `Planetas.c` que resuelva la ecuación de movimiento de los cuerpos del sistema solar. Deben además escribir una rutina

de python llamada `Plots_Planetas.py` que lea los datos producidos por el programa en C y los grafique.

El código `Planetas.c` debe (25 puntos):

- Leer y almacenar las condiciones iniciales de los planetas que se encuentran en el archivo `coordinates.csv`.
- Solucionar la ecuación de movimiento para los planetas usando el método de Leap Frog. Calcule la fuerza sobre cada uno de los planetas debida a la interacción gravitacional con todos los otros y el Sol, así:

$$\vec{F}_i = G \sum_{i \neq j}^N \frac{m_i m_j}{r_{ij}^3} (\vec{r}_j - \vec{r}_i) \quad (2)$$

- Imprimir en un archivo de datos los resultados de la integración numérica de las órbitas.

El código `Plots_Planetas.py` debe (5 puntos):

- Leer y guardar los datos generados por `Planetas.c`
- Hacer una gráfica (guardándola sin mostrarla) de las orbitas de los planetas. Esta gráfica debe luego ser incluida en `Resultados_hw3.pdf`
- Bono (5 pts) Hacer una animación 3D de las orbitas en función del tiempo.

3. (15 points) **Makefile y Resultados.pdf**

En esta tarea las gráficas de resultados se deben presentar en un archivo `Resultados_hw3.pdf` que contenga dos secciones (una para cada punto de la tarea) con títulos y descripciones de las gráficas. Este archivo debe ser generado por `Tarea3.mk` a partir de un archivo `Resultados_hw3.tex`

El archivo `Tarea3.mk` debe:

- Incluir todas las dependencias y reglas necesarias para generar y actualizar el archivo `Resultados_hw3.pdf`.

Los archivos que deben subir a Sicua (comprimidos en `NombreApellido_hw3.zip`) son: `Tarea3.mk`, `Onda.py`, `Planetas.c`, `Plots_Planetas.py` y `Resultados_hw3.tex`. Al correr el makefile se deben generar todas las gráficas, archivos de datos, las animaciones y el pdf.