

Tercera entrega del proyecto

11/12/2015

La tercera y última entrega del proyecto en grupo consiste en la realización de la aplicación final que calcula y dibuja, utilizando Pygame, la posición de distintas masa en el tiempo en una simulación de gravedad.



Aspectos a tener en cuenta

- El programa debe comenzar preguntando al usuario la información de los cuerpos que debe ser cargada. Para ello debe utilizarse la misma estructura que se definió en la entrega anterior. El campo `img`, que no se había utilizado servirá para indicar la imagen asociada a cada cuerpo y que deberá ser dibujada en la posición correspondiente de la pantalla según la posición de los cuerpos en cada momento.

```
# Sistema Tierra - Luna
```

```
nombre:Tierra, masa:5.9722e24, img: imgs/tierra.png, x: 0.0, y: 0.0, fijo: si, vx:0.0, vy:0.0
```

```
nombre:Luna, masa:7.348e22, img: imgs/luna.png, x: 0.0, y: 384402e3, fijo: no, vx: 1023.055, vy: 0.0
```

- En este caso el programa no debe ser capaz solamente de calcular la posición de un único cuerpo en pantalla. Los cálculos de la primera entrega deben ser generalizados para cada cuerpo como le

afecta la gravedad de los demás. Es decir, si en el programa original se calculaba para cada instante la nueva posición de la Luna, segundo a segundo, en este caso se debe realizar el mismo proceso para cada uno de los objetos en pantalla, siendo todos susceptibles de cambiar su posición debida a la interacción con los demás. Así, por cada cuerpo cargado desde fichero, en cada instante de la simulación, se debe calcular su nueva posición a partir de la resultante de fuerzas (suma total) ejercida al calcular la fuerza de debida a cada uno de los otros cuerpos.

- Es necesario que el programa lleva a cabo una transformación de coordenadas entre las coordenadas determinadas en el fichero, que se encuentra en metros y las coordenadas en pantalla, que se indican en píxeles. Para ello se debe tener en cuenta lo siguiente:
 - La posición (0, 0) en metros de la simulación se encontrará en el píxel central de la pantalla.
 - Mediante una regla de tres se determinará a cuantos metros equivalen los W píxeles de anchura y los H de altura, lo que permitirá dibujar otras posiciones en metros en la posición relativa correspondiente en pantalla.
 - El programa en la simulación trabaja siempre en metros para las posiciones y hacer todos los cálculos. Solamente a la hora de dibujar en pantalla se llevan a cabo las transformaciones correspondientes para representar los cuerpos en las posiciones relativas de pantalla.
 - Es necesario implementar, por lo tanto, una función que transforme desde las coordenadas en metros calculadas para cada cuerpo en cada instante a las posiciones en pantalla en píxeles.
- La información anterior de configuración de la pantalla puede leerse desde otro fichero, que debería solicitar el programa al inicio. Por ejemplo, a partir de la información anterior podrían calcularse las posiciones relativas de los cuerpos en pantalla.

```
ancho_pantalla: 800
alto_pantalla: 800
ancho_espacio: 4.9e+11
alto_espacio: 4.9e+11
paso_tiempo: 1500
```

- No es necesario hacer ninguna funcionalidad extra que la que se indica aquí (por ejemplo, aunque la imagen adjunta muestra texto informativo, no es necesario incluir esa funcionalidad en la entrega). Únicamente el pintado básico de las posiciones de los cuerpos de acuerdo a los cálculos realizados y la generación obtenida a N cuerpos.
- Para probar el programa se proporciona la siguiente información, que puede ser utilizada para obtener la simulación del sistema solar mostrada en la imagen. Es necesario convertir los datos al formato que debe usar el programa. Se puede utilizar la aplicación desarrollada para la segunda entrega del proyecto.

```
nombre: Sol
masa: 1.988e30
img: imgs/sol.png
x: 0.0
y: 0.0
fijo: si
```

```
nombre: Venus
```

masa: 4.868e24
img: imgs/venus.png
x: 0.0
y: 1.082e11
fijo: no
vx: -3.502e4
vy: 0.0

nombre: Tierra
masa: 5.9722e24
img: imgs/tierra_peq.png
x: 0.0
y: 1.496e11
fijo: no
vx: -3e4
vy: 0.0

nombre: Marte
masa: 6.419e23
img: imgs/marte.png
x: 0.0
y: 2.28e11
fijo: no
vx: -2.408e4
vy: 0.0