

PLANETARIO TERCERA PRÁCTICA GRÁFICA.

Autores:

Samuel Moreno Vincent

Santiago Rangel Colón



PLANETARIO

Entrega 3 Imagen Digital

Autores:

- Samuel Moreno Vincent
- Santiago Rangel Colón

TABLA DE CONTENIDO

Descripción de la Tarea	2
Menús y Teclas de Funcionamiento:	2
Clases:	3
Foco:	3
Modelo3D:	3
Cámara:	3
setCamara:	3
setFrustrum:	3
encenderCamara:	4
Mundo:	4
Main:	4
Parser:	4
Parser de XML:	5
Metodo de utilizacion:	5
Capturas de Funcionamiento:	6
Captura del sistema:	6
Captura del menú:	6
Captura modo alambre:	7
Captura modo solido:	7
Captura modo FLAT:	8
Captura modo SMOOTH:	8
Captura del sistema sin órbitas:	9



DESCRIPCIÓN DE LA TAREA

Esta ya es la entrega final del planetario, por lo tanto el sistema ya debe tener funcionamiento completo, nuestro planetario ejemplifica al sistema solar, teniendo 9 planetas que giran, en nuestro caso formando círculos alrededor de un astro central, que sería el sol, los planetas pueden tener lunas, que girarían tambien alrededor de sus respectivos planetas, por no sobrecargar demasiado el sistema el máximo de satélites que orbitan en nuestra configuración inicial son 4, pero podrían tener todos los satelites deseados, al mismo tiempo que los astros realizan estos giros, tambien rotan sobre sí mismos, tanto la traslación de los planetas como la rotación de los astros sobre si mismos son en sentido antihorario, tal y como es en la realidad, exceptuando a Venus y Urano que tienen velocidades de rotación negativas ya que son los unícos planetas del sistema solar que giran de forma distinta al resto.

Para poder observar la ruta que siguen los planetas y los satélites de los planetas hemos dibujado lineas por sus rutas formando circulos alrededor, formando sus órbitas.

Además hemos creado puntos de forma aleatoria por la pantalla para hacer el efecto de que sean estrellas que rodean al sistema, esto es simplemente una mejora estética ya que nos parece mucho mas elegante de esta forma.

El sistema comienza en el modo SMOOTH y con el foco 1 (luz blanca) encendido ya que nos parece que es la mejor opción con la que puede comenzar el sistema ya que es la más realista y en la que más tiempo suponemos que estaremos.

MENÚS Y TECLAS DE FUNCIONAMIENTO:

Nuestro sistema tiene un único **menú**, que se abre mediante la pulsacion del **click derecho del ratón**, este menú nos permitirá cambiar entre los distintos modos de vision de los modelos, *Alambre, Sólido*, *FLAT* y *SMOOTH*, tambien nos da la opción *Salir* que terminará la ejecución del sistema. El menú puede observarse en la ilustración 2 en el apartado de capturas.

Los focos serán controlados mediante las teclas 1 a la 8, siendo cada uno de los números los distintos focos que podemos encender o apagar mediante su pulsación, el sistema comienza con el primer foco (luz blanca) encendido cuando ejecutamos el sistema, la tecla 0 es una tecla especial que hemos introducido que apaga todo los focos encendidos.

El cambio entre las cámaras activas se realizará mediante la pulsación de la tecla espaciadora con la que cambiaremos directamente a la siguente cámara de las tres que tenemos creadas inicialmente para nuestro sistema.

Como opción especial hemos introducido la opción de **quitar y poner las órbitas** de los planetas pulsando simplemente la **tecla 9**, de inicio las órbitas estarán encendidas.



Tecla	Acción
Teclas 1 a la 8.	Encendido/Apagado de los Focos.
Tecla 0.	Apagado de todos los focos.
Barra Espaciadora.	Cambio de cámara activa.
Tecla 9.	Poner/Quitar las órbitas de los astros.

CLASES:

FOCO:

Esta clase será la encargada de la creación de cada uno de los 8 focos, estableciendo sus valores de posición, color e intensidad lo cual se hará en el constructor de la misma cogiendo los valores del archivo de configuración xml. Además se podrán cambiar el estado de encendido/apagado hacienda una llamada al método "switchFoco". Los cambios entre los focos activos y no activos se harán mediante la pulsación de las teclas 1 a la 8, siendo cada uno de los números uno de los focos y al pulsar en cada número activaremos o desactivaremos el foco asociado a cada tecla, estas pulsaciones serán escuchadas desde la clase "Main", además podremos apagar todos los focos mediante la pulsación.

MODELO3D:

En la entrega pasada había que ponerle materiales a nuestro modelo, para lo cual creamos un nuevo método llamado "setMaterial", que se encargará de recubrir el material de un material que hemos creado nosotros mismos, además para esta entrega la configuración de los materiales nos entra también como en el caso de los focos por el archivo de configuración xml.

CÁMARA:

Desde la entrega pasada tenemos incluido en nuestro proyecto las cámaras, tenemos solo tres creadas y podemos cambiarlas mediante la pulsación de la barra espaciadora, con la que cambiamos la cámara activa, la clase tendrá 3 métodos:

SETCAMARA:

Este método recibirá por parámetro los 9 parámetros necesarios para el método "glLookAt" y se los asignará a los atributos creados en nuestra clase "Camara" para almacenarlos.

SETFRUSTRUM:

Este método hará lo mismo que el método anterior, guardará en sus atributos correspondientes los parámetros necesarios para realizar el "gluPerspective".



ENCENDERCAMARA:

Este será el método encargado de encender una cámara por lo que se encargará de realizar los métodos "gluPerspective" y "glLookAt" usando como parámetros los atributos guardados en la clase mediante los dos métodos anteriores.

MUNDO:

En esta entrega la clase mundo sigue creciendo, en esta entrega ya no tenemos solo 1 modelo, ya pasamos a tener un sistema solar, por lo tanto ya no vamos a tener solamente 1 modelo, sino una colección de ellos, además ahora la configuración inicial ya no está en el código, pasa a estar en un archivo de configuración con extensión .xml desde donde entrará la información referente al número de modelos y la configuración de estos asi como del resto de componentes de nuestro sistema, esto lo explicaremos más adelante en su sección correspondiente.

En esta entrega además de tener astros estos también tienen que girar alrededor de un astro central, en nuestro caso el Sol, pero además es posible que los astros creados no giren alrededor del sol, sino alrededor de otro astro, estos serán satélites y los diferenciamos gracias a una bandera llamada esLuna que estará en true si este planeta orbita alrededor del astro anterior a él que no sea también un satélite, en el archivo xml de configuración no hemos puesto planetas con más de cuatro satélites porque nos parecían simplemente suficientes, pero puede haber tantos orbitando a un mismo planeta como queramos, la traslación de los astros la hacemos obteniendo el tiempo en milisegundos del sistema mediante la función GLGetElapsedTime.

MAIN:

La funcionalidad de la clase "Main" es la de crear una instancia de mundo y poner en marcha el sistema, además esta clase tiene los menús del sistema, estos menús consisten en un menú que aparece al hacer click derecho del ratón en la interfaz, con este menú podremos cambiar entre los distintos tipos de vistas de nuestro sistema (WIRE, SOLID, FLAT Y SMOOTH), además de eso también tendrá que capturar la pulsación de las teclas que tenemos asignadas a los focos (Tecla 1 a la 8), la pulsación de la barra espaciadora para poder cambiar entre las distintas cámaras existentes, la pulsación de la tecla escape con la que nos saldríamos del sistema y las funciones de zoom y rotación que hacemos mediante el ratón. Mediante la pulsación de la tecla 9 apagaremos todos los focos activos, esto es una modificación persona porque en ocasiones no sabíamos que focos nos habíamos quedado encendidos.

PARSER:

Esta clase se encargará de la gestión de los datos obtenidos del documento XML. En el constructor le pasamos el nodo inicial del árbol de datos.

Los métodos "getNumMateriales", "getNumFocos", "getNumCamaras" y "getNumModelos" obtienen el número de materiales, focos, cámaras y modelos respectivamente, que hay en el árbol de datos haciendo una llamada al método recursivo "getNum recursivo" al cual se pasa por parámetro el inicio del contador



(0), el nodo inicial del árbol de datos, y el nombre del objeto a buscar. Este método hace un recorrido total del árbol y va sumando al contador cada vez que encuentra un objeto con el nombre dado.

Los métodos "getMaterial", "getFoco", "getCamara", "getFrustrum" y "getModelo" obtienen el vector de datos de su respectiva clase con el identificador dado haciendo una llamada al método recursivo "getVector_recursivo" al cual se le pasa el nodo inicial del árbol de datos, el vector que va a ser llenado con los datos, el identificador del objeto, el tipo de vector dentro del objeto, el nombre del objeto y el número de datos que va a obtener. Este devuelve los datos en el vector haciendo un recorrido por todo el árbol de datos hasta encontrar el objeto del tipo especificado y con el identificador dado.

PARSER DE XML:

Las clases xmlnode, xmlnodelist, xmlparser y útil han sido obtenidas de la página: http://jok3n.com/xmlparser/descargas/

Su función es transformar todos los datos dentro del xml en un árbol de nodos. Para ello se realiza una inicialización del parser en la clase Mundo (InitParser) de la cual se obtiene dicho árbol que se guardará en la clase Parser construida por nosotros y explicada anteriormente.

METODO DE UTILIZACION:

Para que el xml sea leído correctamente todos los datos deben tener este formato:

```
<Nombre de la clase + identificador>
<Nombre del vector de datos>
<Lista de datos>
```

Ejemplo:

```
<Modelo id ="0">

<Datos>

<radioGiro>0</radioGiro>

<radioAstro>3</radioAstro>

<velocidad>0.1</velocidad>

<esLuna>0</esLuna>

<velocidadRotacion>0.1</velocidadRotacion>

</Datos>

</Modelo>
```



CAPTURAS DE FUNCIONAMIENTO:

CAPTURA DEL SISTEMA:

Captura de como se ve el sistema al ejecutar la aplicación, tiene activado el foco 1 (luz blanca) y el modo Smooth.

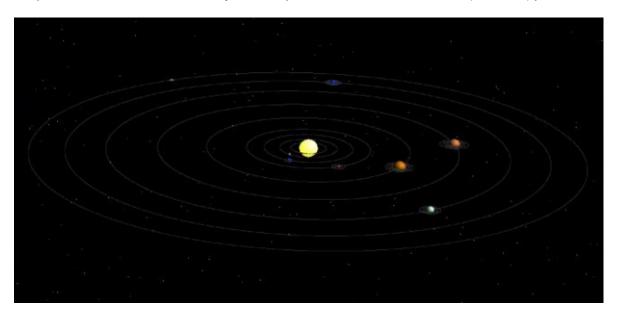


Ilustración 1 Sistema solar en cámara 1 Smooth y foco 1(luz blanca).

CAPTURA DEL MENÚ:

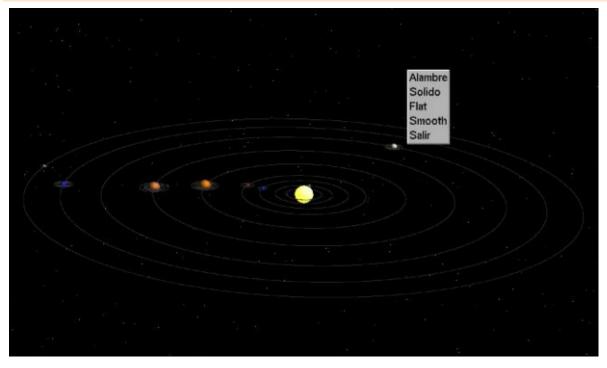


Ilustración 2 Sistema con el menú visible.



CAPTURA MODO ALAMBRE:

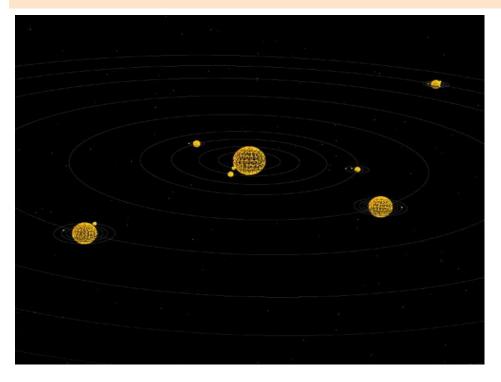


Ilustración 3 Sistema con el modo alambre activo

CAPTURA MODO SOLIDO:

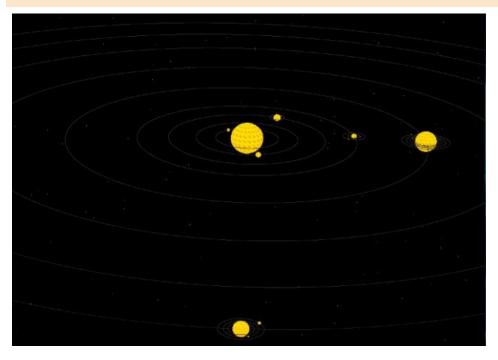


Ilustración 4 Sistema en el modo sólido.



CAPTURA MODO FLAT:

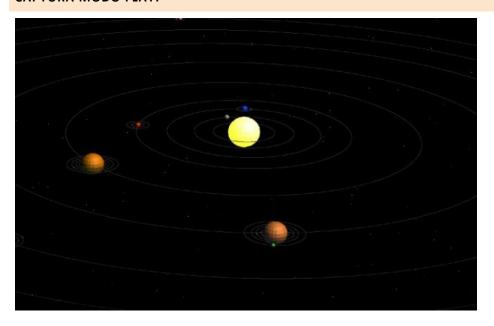


Ilustración 5 Sistema en el modo FLAT.

CAPTURA MODO SMOOTH:

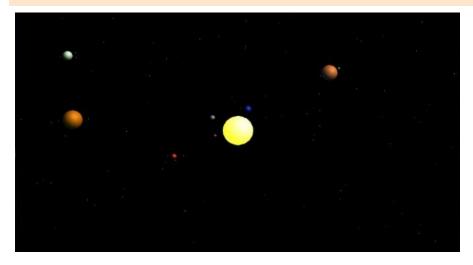


Ilustración 6 Sistema en el modo SMOOTH quitando las órbitas.



CAPTURA DEL SISTEMA SIN ÓRBITAS:



Ilustración 7 Sistema desde lejos con las órbitas desactivadas.