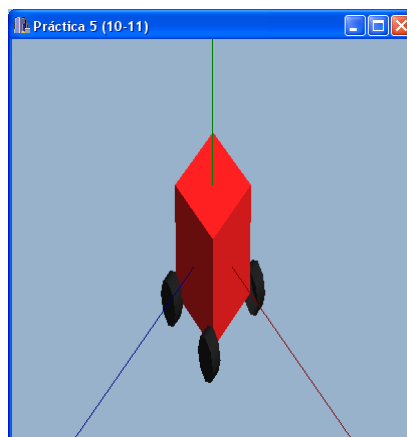
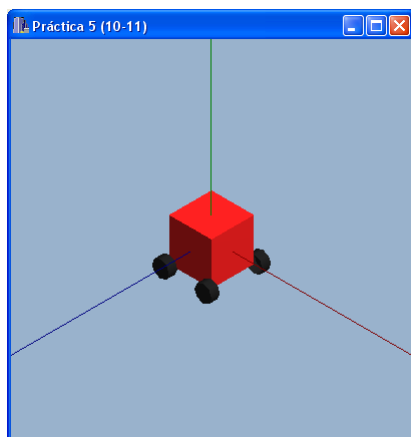


INFORMÁTICA GRÁFICA
Ingeniería en Informática
Curso 2013-2014

PRÁCTICA 6. Versión 2.0. Cámara. Fecha límite de entrega: 30 de abril.

El proyecto debe contener la clase **Camara** de manera que se puedan realizar con ella las siguientes operaciones:

- **roll(α), yaw(α) y pitch(α)**, que giran la cámara, mediante teclado, α radianes, según se explicó en clase. En la **demo1** del Campus Virtual, estas operaciones se realizan con las parejas de teclas u/i, j/k, n/m
- **desplazar(x, y, z)**, que mueve la cámara, mediante teclado, al punto (x, y, z). En la **demo1** aparece el movimiento a lo largo del eje Y con las teclas x, s
- **ortogonal, perspectiva y oblicua**, que muestran la escena de acuerdo con cada una de estas proyecciones. En la **demo1**, las dos primeras proyecciones se obtienen con las teclas o, p, viéndose respectivamente:



- **recorridoEje(x, y, z)** con la que es posible mover la cámara a lo largo de los ejes. En la **demo1**, los pares de teclas (e, r), (d, f), (c, v) permiten recorrer la escena a lo largo de los ejes, yendo de un lado a otro
- **giraX(), giraY(), giraZ()** que permiten ver la escena, dando una vuelta con la cámara alrededor de los ejes X, Y, Z, respectivamente. En la **demo1** esto se lleva a cabo con las teclas 1, 2, 3
- **lateral, frontal, cenital y esquina**, que muestran la escena desde el lado derecho, izquierdo, arriba y punto opuesto al rincón, respectivamente. En la **demo1**, estas cuatro vistas se obtienen con las teclas 4, 5, 6, 7.

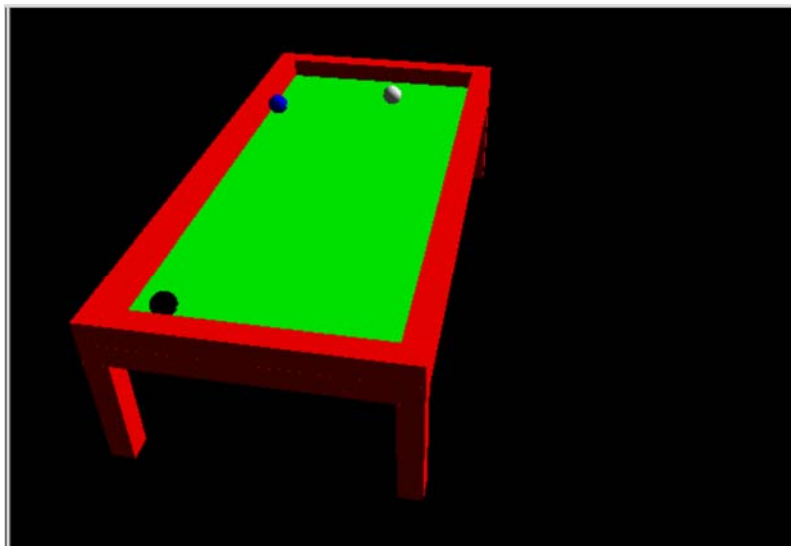
En la **demo1**, las teclas **right, left, up, down, a, z** mantienen las rotaciones de la escena con respecto a los ejes que se tenían en la Práctica 4.

El proyecto debe contener también las siguientes clases:

- **Objeto3D**: es la clase de la que heredan los elementos que intervienen en la escena. Tiene, como atributo, un objeto de la clase **TAfin** que contiene la matriz que define el marco de coordenadas donde se mostrará el objeto.

- **TAfin:** sus objetos tienen, como único atributo, una matriz 4×4 de floats. Esta clase debe definir métodos para calcular las matrices de traslación, escalación y rotación, así como un método para postmultiplicar matrices.
- **Color:** sirve para definir el color de un objeto, mediante combinación de los tres colores primarios.
- **ObjetoCompuesto3D:** hereda de **Objeto3D** y tiene, como atributo, el array de los objetos 3D que lo componen.
- **ObjetoCuadrado:** hereda de **Objeto3D** y tiene, como subclases, **Esfera**, **Cilindro** y **Disco**, cada una de las cuales permite construir el objeto cuadrado correspondiente.
- **Malla:** sirve la clase de la práctica anterior, únicamente es necesario integrarla en el proyecto de forma que herede de la clase **Objeto3D**.
- **Tablero:** hereda de la clase **Malla** y tiene una constructora que permite definir una malla formada por cuadrados (o un entramado formado por cubos) a partir de seis datos que son las tres dimensiones del tablero (largo, ancho y grueso) y el número de partes en que se divide cada una de ellas.

La escena es una mesa de billar. Una primera versión de ella es la que aparece en la imagen, con objetos adicionales que no aparecen, aunque toda ella podrá variar en próximas versiones de esta práctica. La mesa se compone, como ves, de un tablero verde, encuadrado en un marco rojo, sujeto por cuatro patas. Sobre el tablero aparecerán no tres sino quince bolas, una de ellas negra, más una bola blanca, colocadas todas como es habitual al inicio de una partida de billar americano,. También formará parte de la escena un taco –que puede ser un cono truncado- más una tiza –modelable como un cubo con una tapa azul.



La escena entera deberá poder escalarse, trasladarse o rotar, mediante teclado, con respecto a cualquiera de los ejes coordenados. Basta que hagas, cada una de las tres posibilidades, con respecto a un eje solamente.

De momento, son partes opcionales las siguientes:

- Añadir seis troneras a la mesa de billar.
- Programar la desaparición de una bola por una tronera.