

Leed detenidamente el enunciado antes de empezar a hacer nada!

El código que proporcionamos crea y visualiza una escena formada por un suelo que mide 3x3 (XxZ) con el vértice de coordenadas mínimas en el origen de coordenadas, una pelota de diámetro 0.5 y un Patricio de altura 1 los dos con el centro de la base de su caja contenedora en el punto (0,0,0).

Los parámetros de la cámara y las matrices `viewMatrix` y `projectMatrix` están inicializadas en `MyGLWidget` con unos valores por defecto que NO cumplen las especificaciones del enunciado, y la interacción del usuario con el ratón permite modificar interactivamente los ángulos ψ y θ .

En la clase `MyGLWidget` (**que no se puede modificar**) hay un método `creaBuffers` para cada modelo. Estos métodos crean el VAO del modelo y los VBOs necesarios (vértices, material, normales) para poder implementar el cálculo de la iluminación. También proporcionamos las rutinas `Lambert` y `Phong` que se encuentran en el Vertex Shader.

Observación-1: Analizad las clases `MyGLWidget` y `NouGLWidget` y los shaders ANTES de implementar los ejercicios requeridos.

Observación-2: RECORDAD QUE TODOS los métodos que teneis que crear o modificar SON DE LA CLASE `NouGLWidget`.

En la valoración del ejercicio 6 tendrá mucha importancia el diseño y la usabilidad de la interfaz.

1. (1.5 puntos) Modifica la escena para que el suelo mida 6x9 (XxZ) sin modificar su vértice de coordenadas mínimas que debe continuar siendo (0,0,0); que haya un Patricio de altura 2 (escalado uniformemente) situado con el centro de la base de su caja contenedora en el punto (3,0,8) y mirando en dirección Z-; y una pelota de diámetro 0.5 con el centro de la base de su caja contenedora en el punto (3,0,7). La orientación original del Patricio es mirando hacia Z+.

Para hacer este ejercicio debes modificar los métodos `modelTransformPatricio`, `modelTransformPilota` y `modelTransformTerra`.

2. (1 punto) Modifica las matrices `viewMatrix` y `projectMatrix` para obtener una visión de la escena en cámara en tercera persona, de forma que la escena esté centrada en el viewport, ocupando el máximo de éste y que no se deforme ni se recorte inicialmente (puede recortar si se modifica la ventana OpenGL). La orientación inicial de la cámara será con los ángulos $\psi = 0$ y $\theta = \pi/6.0$.

Concretamente hay que:

- Modificar el método `iniEscena` para inicializar correctamente los atributos de escena necesarios.
 - Modificar los métodos `iniCamera`, `viewTransform` y `projectTransform` para calcular correctamente las matrices `viewMatrix` y `projectMatrix`.
3. (1 punto) Añade a la escena el cálculo de la iluminación usando el modelo de iluminación de Phong calculado en el Vertex Shader. Se pide un foco de luz de escena blanco en la posición (3,4,8).
 4. (2.5 puntos) Implementa el método `keyPressEvent` para que mediante las teclas 'W' y 'S' la pelota se mueva en la dirección del eje Z. Si la pelota sale del suelo debe desaparecer de la escena (no se pinta) hasta que vuelva a estar de nuevo sobre el suelo, mientras que si topa con el Patricio debe detenerse.

Concretamente:

- Tecla 'W' - pelota se traslada 0.5 en dirección Z-
- Tecla 'S' - pelota se traslada 0.5 en dirección Z+

En el momento que la pelota desaparece del suelo, se debe cambiar el color del foco de luz para que sea un foc amarillo, y debe quedarse amarillo hasta que la pelota vuelva a aparecer en la escena, que volverá a ser blanco.

Consideramos que la pelota topa con el Patricio cuando el centro de la base de su caja tiene coordenada $Z=7$.

Consideramos que la pelota sale del suelo si el centro de la base de su caja está fuera de los límites del suelo.

5. (2 puntos) Añade una nueva cámara de vista en planta (vista desde arriba). Esta cámara debe ser una cámara ortogonal que permita ver toda la escena entera, centrada y ocupando el máximo del viewport. Esta cámara no deformará en caso de redimensionamiento de la ventana OpenGL. Para cambiar entre las dos cámaras usaremos la tecla 'C'.

Concretamente hay que:

- Modificar los métodos `iniCamera`, `viewTransform` y `projectTransform` para añadir las inicializaciones que sean necesarias y el cálculo de las matrices `viewMatrix` y `projectMatrix` también en el caso de la nueva cámara.
 - Implementar el tratamiento de la tecla 'C' en el método `keyPressEvent` de forma que éste haga el intercambio entre las dos cámaras existentes.
6. (2 puntos) Añade a la interfaz los elementos adecuados para que el usuario pueda realizar el movimiento de la pelota y también el cambio de cámara. Estos elementos de interfaz deberán actualizarse también adecuadamente, si es el caso, cuando el usuario interactue directamente con las teclas.