

**Leed detenidamente las instrucciones y el enunciado
antes de empezar a hacer nada!**

Instrucciones

1. Podéis usar el código que habéis elaborado en las clases de laboratorio y que tengáis en vuestra cuenta, pero **sólo el código que hayais generado vosotros**; no podéis usar código que otros estudiantes hayan compartido con vosotros (ni que hayais compartido con otros estudiantes). De lo contrario se considerará copia.
2. Partiremos del código que tenéis en `examen.tgz` (adjunto a esta práctica). Tenéis que descomprimir este archivo en un directorio vuestro. Os creará un subdirectorio `examen` donde tendréis todos los ficheros con los que tenéis que trabajar. **No tenéis que modificar los ficheros `examen.pro` ni el `main.cpp`**. Los ejercicios que os pedimos sólo requieren cambios en la clase `MyGLWidget` y en los shaders.
3. **Si el código que entreguéis no compila o da error de ejecución, la evaluación será un 0**, sin excepciones.
4. Para hacer la entrega tenéis que generar un archivo tar que incluya todo el código de vuestro examen que se llame `<nombre-usuari>_GL.tgz`, donde sustituiréis `<nombre-usuari>` por vuestro nombre de usuario. Por ejemplo, el estudiante Pompeu Fabra (desde una terminal en la que se ha colocado dentro del directorio `examen`):

```
make distclean
tar zcvf pompeu.fabra_GL.tgz *
```

Es importante el `'make distclean'` para borrar los archivos binarios generados; que el nombre de usuario sea el correcto (el vuestro); y que tenga la raya baja `'_'` separando el nombre de usuario del sufijo `GL.tgz`

5. Una vez hecho esto, en vuestro directorio `examen` tendréis el archivo `<nombre-usuario>_GL.tgz` que es lo que tenéis que entregar. **Haced la comprobación**, descomprimiendo este archivo **en un directorio completamente vacío**, que el código que entreguéis compila (haciendo `qmake; make`) y ejecuta correctamente.
6. Finalmente, entregad el fichero en <https://examens.fib.upc.edu>

Nota: Recordad que si abríis el fichero `/assig/idi/man_3.3/index.html` desde el Firefox o el Konqueror tendréis acceso a las páginas del manual de OpenGL 3.3, y con el fichero `/usr/include/glm/doc/api/index.html` tendréis acceso a las páginas del manual de la librería glm. También tenéis, como ya sabéis, el `assistant` para dudas de Qt.

Enunciado

El código que os pasamos pinta un suelo de 20x20 centrado en el origen y un Patricio de altura 1 con el centro de su base en el punto (0, 0, 2.5) y con una cámara inicializada de forma arbitraria (ver una imagen en el fichero `EscIni-30.png`). Tiene inicializados todos los datos de materiales y normales necesarios para poder implementar el cálculo de la iluminación. También os pasamos los métodos `Lambert` y `Phong` que se encuentran en el Vertex Shader.

1. (3 puntos) Modifica esta escena para que, en lugar de un Patricio de altura 1 y con el centro de su base en el punto $(0, 0, 2.5)$, haya 4 Patricios de altura 2 con sus bases centradas respectivamente en los puntos $(8, 0, 8)$, $(8, 0, -8)$, $(-8, 0, -8)$ i $(-8, 0, 8)$. Estos 4 Patricios deben estar todos mirando hacia el centro, es decir hacia el eje Y de la escena.

Esta escena se debe ver centrada y sin recortar, y aprovechando el máximo del viewport (vista), con una cámara perspectiva. En caso de redimensionamiento de la ventana (resize) la escena no debe recortarse ni deformarse.

Una imagen posible de la solución a este ejercicio la podéis ver en el archivo `EscSo11-30.png`.

2. (2 puntos) Añade a la escena el cálculo de la iluminación **en el Vertex Shader** usando el modelo de Lambert y con un foco de luz blanca en la posición $(5, 3, 5)$ de la escena. Modifica también las propiedades del material del suelo para que sea de un material amarillo mate.
3. (2 puntos) Haz que con las teclas 'W' y 'S' los 4 Patricios se vayan moviendo de forma que su base se acerque o se aleje del centro del suelo, de forma que con 'W' todos se mueven acercándose hacia el centro y con 'S' todos se mueven alejándose del centro (en la misma dirección que cuando se acercan pero en sentido contrario). En el caso de que pulsando la tecla 'W' (es decir intentando ir hacia el centro) la posición a la que les tocaría ir está a distancia inferior a 1 del centro, no se moverán más en esta dirección (es decir 'W' ya no tiene ningún efecto hasta que volvamos a pulsar 'S').
4. (2 puntos) Haz que al pulsar la tecla 'C' se cambie de cámara de forma que ahora pasamos a tener una cámara justo encima del primer Patricio (el que está inicialmente en la posición $(8, 0, 8)$) a una altura de 2.5 y mirando en la misma dirección en la que mira el Patricio. Esta segunda cámara debe ser una cámara perspectiva con ángulo de apertura de $M_PI/2.0$ radianes (90 grados) y Znear y Zfar adecuados para ver todo lo que el Patricio vería de la escena desde su posición. La cámara se debe mover de la misma forma que el Patricio y no debería deformar la escena en caso de redimensionamiento del viewport.

Pulsando de nuevo la tecla 'C' se debe poder recuperar la cámara inicial o anterior.

Para la posición y orientación de esta segunda cámara puedes usar tanto la llamada lookAt como transformaciones geométricas, lo que prefieras.

5. (1 punto) Haz que al pulsar la tecla 'L' el foco de luz pase a ser cian. Si se vuelve a pulsar 'L' vuelve a ser blanco y así sucesivamente.