

Examen 2025

Pregunta 1. Tipo test (3 puntos)

12 preguntas (muchas con respuesta **B**):

- Mandar vértices a la GPU (`VAO` , `VBO` , `GL_TEXTURE` , `EBO`)
 - Qué decide qué caras se renderizan y cuáles no (`GL_CULL_FACE` , `Blending test` , `alpha test`)
 - Dónde se transforman los vértices (`Vertex shader` , `Fragment shader`)
 - Diferencia entre `DisplayList` y ...
 - Diferencia entre `glDrawElements` y `glDrawArrays`
 - Una cuya respuesta es el `Fragment shader`
-

Pregunta 2. (3.5 puntos)

Transformaciones OpenGL:

```
glLoadMatrix(...);
glPushMatrix();
glTranslatef(2, 0, 0);
glRotatef(45, 0, 0, 1);
glScalef(1.5, 1.5, 1.5);

glBegin(GL_QUADS);
    -1, -1
    1, -1
    1, 1
    -1, 1
glEnd();
```

Pregunta 2.1 (1 punto)

- ¿Cómo se aplican las matrices de transformación en OpenGL y cómo afecta esto al resultado final?

- ¿Qué tendríamos que cambiar en el código anterior para rotar el cuadrado sobre su centro y después trasladarlo?
(Suponer que está en `(2.0, 2.0, 0.0)`)
-

Pregunta 2.2 (1.5 puntos)

- Dame la matriz de transformación final (en orden matemático correcto) y calcula la posición final del punto `(-1, -1, 0)`.
(Dado: `cos(45°) = sin(45°) = 0.707`)
-

Pregunta 2.3 (1 punto)

- Dame el código para programar las transformaciones en OpenGL 3.3 (con `glm` y shaders)
-

Pregunta 3. (3.5 puntos)

Sobre **iluminación**:

Se dan los vértices del mismo cuadrado de la pregunta 2.

- Luz direccional: `(0, 0, 1)`
 - Intensidad ambiental: `(0.2, 0.2, 0.2)`
 - Intensidad difusa: `(0.7, 0.7, 0.7)`
 - Coef. reflexión ambiental: `(0.1, 0.1, 0.1)`
 - Coef. reflexión difusa: `(1, 1, 1)`
-

Pregunta 3.1 (1 punto)

- ¿Cuál es la normal de la cara?
- ¿Afectaría un escalado uniforme a la normal? ¿Y un escalado no uniforme?
- Si tuviéramos un cubo con esta cara y se viera todo uniforme (todas las caras se ven iguales),
¿cuál es el error más probable y cómo arreglarlo?

Pregunta 3.2 (1.5 puntos)

- Calcula el color en el punto central de la cara mediante los modelos de sombreado de **Gouraud** y **Phong**.
Compara los resultados y razona.

Pregunta 3.3 (1 punto)

- Haz el **vertex shader** y **fragment shader** para lo anterior.