

TEXTURAS.pdf



flowerpower22



Informática Gráfica



3º Grado en Ingeniería Informática



Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación
Universidad de Granada



**Todas tenemos una amiga
experta en recorrer
kilómetros en discotecas.
Y si no la tienes eres tú.**

Para ser una experta en kilómetros
de verdad, déjate guiar por **coches.net**



Compra
o vende
tu coche

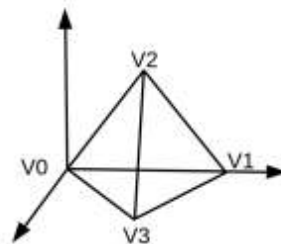
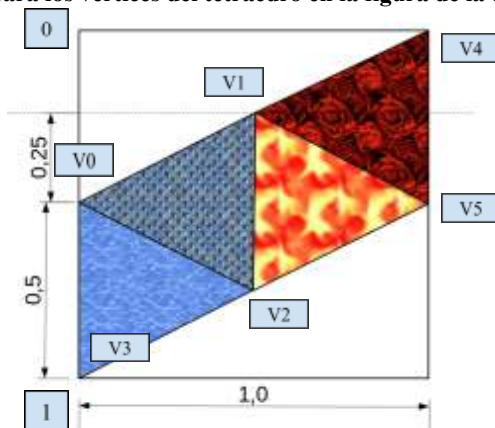
✓ fácil
✓ rápido

coches.net



Tu colega "el experto en Erasmus": se fue pensando en aprobarlo todo.
No aprobó nada. Lo probó todo.

Escribe el código en C++ con OpenGL que dibuje un tetraedro (no necesariamente regular) aplicándole la textura que se muestra en la figura de la izquierda. Anotar las coordenadas usadas para los vértices del tetraedro en la figura de la derecha. Distancia $V_0-V_1 = 1$



Definimos las estructuras:

```
vector<_vertex3f> Vertices;
vector<_vertex3ui> Triangles;
vector<_vertex2f> Textures;
```

```
Vertices.resize(4); // definimos los vértices
Vertices[0].x=_vertex3f(0,0,0);
Vertices[1].x=_vertex3f(1,0,0);
Vertices[2].x=_vertex3f(0.5,1,0.5);
Vertices[3].x=_vertex3f(0.5,0,1);
Triangles.resize(8); // definimos las caras
Triangles[0] = _vertex3ui(0,3,2);
Triangles[1] = _vertex3ui(3,0,2); // cara izquierda
Triangles[2] = _vertex3ui(3,1,2);
Triangles[3] = _vertex3ui(1,2,3); // cara derecha
Triangles[4] = _vertex3ui(1,0,2);
Triangles[5] = _vertex3ui(0,1,2); // cara trasera
Triangles[6] = _vertex3ui(0,1,3);
Triangles[7] = _vertex3ui(1,0,1); // cara inferior
Textures.resize(8); // definimos las texturas
Textures[0] = _vertex2f(0,0.5);
Textures[1] = _vertex2f(0.5,0.25);
Textures[2] = _vertex2f(0.5,0.75);
Textures[3] = _vertex2f(1,1);
Textures[4] = _vertex2f(1,0);
Textures[5] = _vertex2f(1,0.5);
glBegin(GL_TRIANGLES);
for(int i = 0; i < Triangles.size(); i++){
    glTexCoord2f((GLfloat *) &Textures[Triangles[i]._0]);
    glVertex3f((GLfloat *) &Vertices[Triangles[i]._0]);
    glTexCoord2f((GLfloat *) &Textures[Triangles[i]._1]);
    glVertex3f((GLfloat *) &Vertices[Triangles[i]._1]);
    glTexCoord2f((GLfloat *) &Textures[Triangles[i]._2]);
    glVertex3f((GLfloat *) &Vertices[Triangles[i]._2]);
}
glEnd();
```

