
Introducció

En aquests exercicis us demanen que implementeu plugins (en C++) pel visualitzador bàsic de l'assignatura. Alguns exercicis requereixen escriure shaders (en GLSL).

Farem servir aquesta nomenclatura:

- **Effect plugin:** plugin que implementa els mètodes **preFrame** i/o **postFrame** (a banda d'altres mètodes que pugueu necessitar). Teniu un exemple a **plugins/effectcrt**.
- **Draw plugin:** plugin que implementa el mètode **drawScene** (a banda d'altres mètodes que pugueu necessitar). Teniu un exemple a **plugins/drawvbong**.
- **Action plugin:** plugin que implementa mètodes d'entrada com ara **keyPressEvent** (a banda d'altres mètodes que pugueu necessitar). Teniu un exemple a **plugins/navigatedefault**
- **Render plugin:** plugin que implementa el mètode **paintGL**, (a banda d'altres mètodes que pugueu necessitar). Teniu un exemple a **plugins/renderdefault**. Per desenvolupar shaders a la primera meitat del curs heu fet servir el plugin que podeu trobar a **plugin/shaderLoader/**
- VS, GS i FS fan referència a vertex shader, geometry shader i fragment shader, resp.

Warming up

Desplega les fonts del visualitzador (viewer) en un directori teu i prova construir els binaris, seguint les instruccions que us hem donat. Executa el viewer i prova de carregar algun model (en format .obj).

ModellInfo (1)

Escriu un **Effect Plugin** que escrigui (**postFrame**) la següent informació sobre l'escena: número total d'objectes carregats, número total de polígons, número total de vèrtexs, i el percentatge de polígons que són triangles. Feu una implementació que tingui un impacte negligible en el frame rate. Teniu un exemple de recorregut de l'escena a **plugins/drawvbong**

En aquesta versió només cal mostrar aquests valors pel canal de sortida estàndard (cout...).