

# Interacció i disseny d'interfícies

## Examen OpenGL 2020-21 / Q2

Dept. de Ciències de la Computació

Professors INDI

### 1 Instruccions

1. Podeu usar el codi que heu elaborat en les classes de laboratori i que tingueu al vostre compte, però **sols el codi que hagueu generat vosaltres**; no podeu fer servir codi que altres estudiants hagin compartit amb vosaltres (ni que hagueu compartit amb d'altres estudiants). Altrament correu el risc que es pugui considerar el vostre examen com una còpia.
2. Partirem del codi que teniu a `examen.tgz` (adjunt a aquesta tasca). Heu de desplegar aquest arxiu en un directori vostre. Us crearà un subdirectori `examen` on tindreu tots els fitxers amb els que heu de treballar. Els exercicis que es demanen només requereixen canvis a la classe `MyGLWidget`, als shaders i al fitxer `MyForm.ui` usant el `designer`. **No cal modificar cap altre fitxer dels que us donem.**
3. Per a fer el lliurament heu de generar un arxiu TGZ que inclogui tot el codi del vostre examen que es digui `INDI_ex_opengl_DNI.tgz`, on substituïreu `DNI` pel vostre DNI incloent la lletra. Per exemple, l'estudiant amb dni `11111111H` (des d'una terminal en la que s'ha col·locat dins del directori `examen`):

```
make distclean
tar cvzf INDI_ex_opengl_11111111H.tgz *
```

És important el `'make distclean'` per a esborrar els arxius binaris generats; que el DNI sigui el correcte (el vostre); i que hi hagi el sufix `.tgz`.

4. **Si el vostre lliurament no segueix el format indicat, està comprimit usant una altra eina, no compila o dóna error d'execució, l'avaluació serà un 0 sense excepció.**
5. Un cop fet això, al vostre directori `examen` tindreu l'arxiu `INDI_ex_opengl_DNI.tgz` que és el que heu de lliurar. **Feu la comprovació**, descomprimint aquest arxiu **en un directori completament buit**, que el codi que lliureu compila (fent `qmake`; `make`) i executa correctament.
6. Finalment, lliureu el fitxer a <https://atenea.upc.edu> mitjançant la tasca "Prova OpenGL".

## 2 Enunciat

El codi que proporcionem crea i visualitza una escena formada per:

- un terra de 30x20 unitats ubicat sobre el pla XZ amb una punta tocant l'origen de coordenades.
- un cub de color marró de costat 0,5 amb el centre de la seva base al punt (1, 0, 0),
- un cub de color gris de costat 0,5 amb el centre de la seva base al punt (2, 0, 0)
- un cotxe situat amb el centre de la seva base a l'origen de coordenades.

La càmera està inicialitzada arbitràriament i només es pot modificar interactivament l'angle  $\psi$ .

Hi ha un mètode *creaBuffers* per a cada model. Cdascun d'aquests mètodes té inicialitzades totes les dades de material i normals necessàries per poder implementar el càlcul de la il·luminació. També es proporcionen les rutines *Lambert* i *Phong* que es troben al Vertex Shader.

**Observació: Analitzeu el codi donat abans d'implementar funcionalitats.**

1. (1,5 punts) Modifica l'escena per a que, el cotxe tingui alçada 2 (escalat uniformement) i estigui situat amb el centre de la base de la seva capsa contenidora al punts (0, 0, 0) mirant en direcció X+. Observació: el model `porsche.obj` inicialment mira en direcció Z+.

Primerament hauràs d'utilitzar el model del cub1 per a construir tres parets que volen simular les parets d'un pàrking: Paret1, Paret2 i Paret3. La Paret1 ha de tenir mides 0,5x5x10 (mida en X, Y i Z respectivament) i el centre de la seva base al punt (-15,0,0). La Paret2 ha de tenir mides 10x5x0,5 i el centre de la seva base al punt (-10,0,5). I la Paret3 ha de tenir mides 10x5x0,5 i el centre de la seva base al punt (-10,0,-5).

També hauràs d'utilitzar el model del cub2 per a construir una porta per les parets. Aquesta porta ha de tenir mides 0,3x5x10 (mida en X, Y i Z respectivament) i situada amb el centre de la seva base al punt (-5, 0, 0).





2. (1,5 punts) Implementa correctament el càlcul de la càmera (`viewMatrix` i `projectMatrix`) per tenir una càmera perspectiva en tercera persona de manera que l'escena es vegi en tot moment sencera, centrada, sense deformació i ocupant el màxim del viewport, encara que l'usuari modifiqui la finestra gràfica.



En cas de redimensionament de la finestra (resize) l'escena no s'ha de deformar.



Aquesta càmera també ha de permetre la inspecció mitjançant rotacions dels angles d'Euler (angles  $\psi$  i  $\theta$ ), és a dir, l'usuari ha de poder modificar aquests angles utilitzant el ratolí com s'ha fet al laboratori.

La càmera inicial ha de tenir angles  $\psi = 0.5$  i  $\theta = 0.5$ . Una imatge de la solució a aquests 2 primers exercicis la tens a [EscSol1.png](#).

3. (1.5 punts) Modifica adientment el **Vertex Shader** i el **Fragment Shader** per afegir a l'escena el càlcul d'il·luminació usant el model d'il·luminació de Phong calculat al **Fragment Shader**. El focus de llum és inicialment blanc i de càmera, situat exactament a la posició de la càmera. La posició del focus es passa com a uniform als shaders.






Fes que prement la tecla  (Key\_F) el focus canviï a un focus d'escena situat en tot moment a sobre del cotxe (la posició inicial d'aquest focus si el cotxe no s'ha mogut seria (0, 15, 0)). El primer cop que es prem la tecla  el focus passa a ser d'escena, i si es torna a prémer la tecla , el focus torna a ser l'inicial de càmera. El comportament de la tecla  és cíclic.

4. (1,5 punts) Fes que la porta s'elevi de manera que pugui entrar el cotxe en el parking. Prement la tecla  (Key\_U) la porta s'eleva una unitat fins un màxim de 5 i prement la tecla  (Key\_D) la porta baixa una unitat fins arribar al terra. Una imatge on es pot veure l'escena amb la porta del pàrking oberta la tens a [EscSol2.png](#).

5. (2 punts) Afegeix la possibilitat que el cotxe vagi endavant i endarrera. En concret, quan es premi  (Key\_Up) ha d'incrementar la posició del cotxe en una unitat en l'eix de les X; i quan es premi  (Key\_Down) ha de decrementar la posició del cotxe en una unitat en l'eix de les X.

IMPORTANT: Cal controlar que el cotxe no surti fora dels límits del terra i tampoc travessi la porta si no està completament oberta.

6. (2 punts) Afegeix a la interfície els següents elements d'interfície que permetin:

- (a) Decidir quin focus es vol tenir activat en cada moment. Aquest element ha d'estar coordinat amb l'efecte de la tecla .
- (b) Obrir i tancar la porta. Aquest element s'ha d'actualitzar correctament amb l'efecte de les tecles  i .
- (c) Controlar el moviment del cotxe. Aquest element s'ha d'actualitzar correctament amb l'efecte de les tecles  i .

7. (1 punt) **Exercici extra** Afegeix la possibilitat de fer un "reset" (reinici) de tot el comportament de l'aplicació:

- Posar el cotxe a la seva posició inicial.
- Posar activa la càmera amb els valors inicials de la càmera en tercera persona de l'exercici 2.
- Posar el focus com a focus de càmera.

Aquest "reset" es farà mitjançant la tecla  (Key\_I) i ha de reiniciar també els elements d'interfície adientment.