

## Exercici lliurable 2 de laboratori d'IDI 2023–2024 Q2

### Instruccions

1. Aquests exercicis són individuals, així que només pots lliurar **codi que hagi generat tu**. No pots fer servir codi que altres estudiants hagin compartit amb tu ni que tu hagi compartit amb d'altres estudiants. Altrament es considerarà còpia.
2. Partiràs del codi que tens a Exercici-2.tgz (el podeu trobar en el Campus digital). Has de descomprimir aquest arxiu en un directori. Es creara un subdirectori exercici2 on tindras tots els fitxers amb els que has de treballar.
3. La solució que lliuris ha de compilar i executar correctament al laboratori. Si no compila o dóna error d'execució, l'avaluació de l'exercici serà un 0, sense excepció.
4. Els exercicis que es demanen només requereixen canvis a la classe `MyGLWidget`. No has de modificar cap altre fitxer dels que se't proporcionen, ni tampoc canviar el seu nom. A més, els mètodes `iCarregaShaders`, `creaBuffersTerra` i `creaBuffersModels` de la classe `MyGLWidget` no els pots modificar.
5. Per fer el lliurament has de generar un arxiu TGZ que inclogui tot el codi del teu exercici i que es digui `INDI_exercici2_<DNI>.tgz`, on substituiràs `<DNI>` pel teu DNI amb lletra majúscula i sense guions. Per exemple, l'estudiant amb DNI 12345678Z (des d'una terminal en la que s'ha col·locat dins del directori exercici2) farà:

```
make distclean
tar zcvf INDI_exercici2_12345678Z.tgz *
```

és important el 'make distclean' per a esborrar els arxius binaris del directori, que el DNI sigui el correcte (el teu), i que hi hagi el sufix .tgz.

6. Un cop fet això, al teu directori tindràs l'arxiu `INDI_exercici2_<DNI>.tgz` que és el que has de lliurar al campus virtual **abans del dilluns dia 13 de maig a les 23:55**.

## Enunciat

El codi que proporcionem pinta una escena amb un terra de 5x5 centrat a l'origen, un model de Luke Skywalker i un model d'un arbre amb els centres de les seves tapes inferiors a l'origen. Es donen ja implementats els mètodes que construeixen els VAOs i VBOs dels tres models (`creaBuffersTerra()`, `creaBuffersModels()`). La càmera que es dóna a l'esquelet es una càmera completament arbitrària que permet veure l'escena inicialment, es pot veure una imatge de com queda l'escena inicialment a la figura 1.

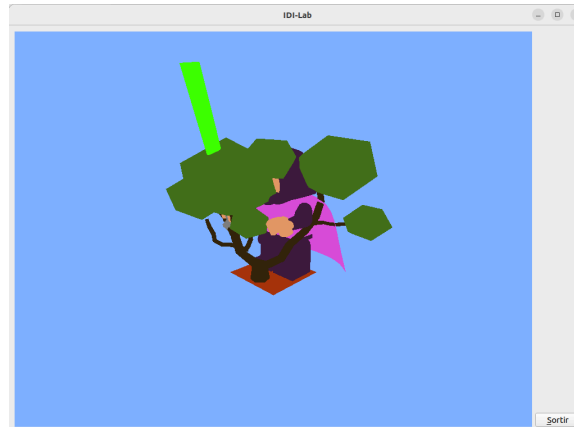


figura 1

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

Es demana que, donat l'esquelet inicial que us hem proporcionat, resolgueu els següents exercicis:

1. Modifica l'escena per a què hi hagi inicialment 5 arbres col·locats de manera aleatòria sobre el terreny escalats de manera uniforme també amb mides aleatòries tenint en compte que l'alçada dels arbres ha d'estar entre 0.08 i 0.13. A més, cada arbre pot estar girat sobre el seu eix vertical un número de graus aleatori. El Luke ha de trobar-se inicialment a la posició (0,0,0) i tenir alçada 1. Heu de crear la funció `setTrees(int numTrees)`, que defineix el nombre d'arbres mostrats a l'escena per a que sigui `numTrees`. La funció calcula i emmagatzema la posició, escala i rotació dels `numTrees` arbres sobre el terreny, i demana a OpenGL que redibuixi l'escena. Heu d'afegir un element d'interfície que permeti seleccionar quants arbres es generaran a l'aplicació (de 5 a 30) i que l'escena s'actualitzi quan aquest número canviï.

Fixa't que l'esquelet té un mètode per a calcular la capsa contenidora de qualsevol model.

La imatge inicial de l'aplicació ha de ser la que mostra la imatge de la figura 2.



figura 2

2. L'escena s'ha de poder visualitzar amb una càmera en tercera persona que permeti inicialment veure l'escena centrada, sencera, sense deformar i ocupant el màxim del viewport (essent el viewport tota la finestra gràfica). La càmera ha de tenir una òptica perspectiva. En cas de redimensionament de la finestra (resize) l'escena no s'ha de deformar ni retallar.

Aquesta camera 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  $\psi = 0$  i  $\theta = -M\_PI/4.0$ . El moviment d'inspecció que es demana ha de fer que en moure l'usuari el ratolí cap a la dreta del viewport la càmera es mou justament en sentit contrari, cap a l'esquerra, i el mateix anant cap amunt. Fixa't que amb aquest moviment d'inspecció sembla que l'usuari arrossegui l'escena en moure el ratolí.

3. Implementa el moviment del Luke de manera que:

- Mitjançant la tecla *Left* (*Qt::Key\_Left*) el Luke rotarà  $M\_PI/4$  cap a la seva esquerra al voltant del seu eix vertical.
- Mitjançant la tecla *Right* (*Qt::Key\_Right*) el Luke rotarà  $M\_PI/4$  cap a la seva dreta al voltant del seu eix vertical.
- Mitjançant la tecla *Up* (*Qt::Key\_Up*) el Luke avançarà en la direcció on està mirant tenint en compte que no podrà sortir mai del terreny.

4. Implementa una segona càmera que s'activi i es desactivi mitjançant la tecla *C* (*Qt::Key\_C*). Aquesta nova càmera estarà situada sobre l'eix *Y* de l'escena i a 2 vegades el radi de l'esfera contenidora i haurà de mirar sempre al centre del terra. Serà una càmera ortogonal i amb *zNear* i *zFar* suficients per a que no es retalli l'escena. La mida de la finestra de projecció de la càmera està ajustada a l'esfera contenidora.

El resultat d'activar aquesta càmera es pot veure a la figura 3. Quan estigui activada aquesta càmera les interaccions amb el ratolí no han de modificar els angles d'Euler de l'altra càmera. A més has d'afegir un element d'interfície que permeti canviar entre les dues càmeres.



figura 3

5. Implementa el reset de l'escena. Mitjançant la tecla *R* (*Qt::Key\_R*) l'escena ha de tornar al seu estat inicial (el que es pot veure a la figura 2). A més has d'afegir un element d'interfície que faci la mateixa acció que aquesta tecla.

6. (Exercici extra) Afegir una funcionalitat més que permeti, amb les tecles '+' i '-' girar tots els arbres al voltant del seu eix vertical i un element d'interfície sincronitzat que permeti controlar-ho.

**IMPORTANT:** Les accions que es realitzin amb el teclat han d'estar coordinades amb l'element d'interfície corresponent.

A l'Atenea tens un executable de la solució demanada.