

<! --

# Prova d'avaluació continuada 1

—

## Matemàtiques i física per a la Multimèdia

-->



@ Aitor Javier Santaeugenia Marí

04/10/2015

## INDEX

[Abans de començar – Pàgina 3](#)

[Apartat A d'exercicis – Pàgina 3](#)

[Exercici 1 – Pàgina 3](#)

[Exercici 2 – Pàgina 4](#)

[Exercici 3 - Pàgina 5](#)

[Exercici 4 - Pàgina 5](#)

[Exercici 5 - Pàgina 6](#)

[Exercici 6 - Pàgina 6](#)

[Exercici 7 - Pàgina 6](#)

[Exercici 8 - Pàgina 6](#)

[Apartat B d'exercicis - Pàgina 7](#)

[Conclusió - Pàgina 7](#)

[Bibliografia - Pàgina 8](#)

### Abans de començar

En el codi de cada exercici hem introduït:

```
//-----  
//-----  
// CANVI  
//-----  
//-----
```

Per tal de deixar més clar **on hem modificat i què hem modificat**. Al mateix temps, també hem senyalat on hi ha la càmera, l'inici de l'escena, la renderització, ja que amb tantes línies de codi, ens tornaven boixos.

<pre>///////////////// // CAMERA // /////////////////</pre>	<pre>///////////////// // SCENE // /////////////////</pre>	<pre>///////////////// // RENDERER // /////////////////</pre>
I altres		

En la documentació, afegim la línia del que hem modificat, per tal de tenir una referència al llegir el document sobre on està la modificació i així facilitar la cerca.

(A) - Proposar una modificació per cadascun d'aquests 8 primers exemples explicant que s'ha modificat i perquè

#### Exercici 1 – El teu primer polígon

Hem fet diverses proves amb la càmera, hem creat una càmera ortogonal (línia 114, no implantada ja que no serviria per veure el resultat, ho intentarem amb un altre exercici, si eliminem “//” la podem veure com queda, però per l'estil de l'exercici no resulta útil).

- Hem creat un altre “shape” (no un polígon), en aquest cas un cercle. El codi emprat és (línia 201):

```
var material = new THREE.MeshBasicMaterial({  
    color: 0x552236,  
    side:THREE.DoubleSide  
});  
  
var radius = 1;  
var segments = 32;  
  
var circleGeometry = new THREE.CircleGeometry( radius, segments );  
var circle = new THREE.Mesh( circleGeometry, material );  
//Deixarem el cercle al mig del quadrat i triangle  
circle.position.set(0.0, 0.0, 1.0);  
scene.add( circle );  
}
```

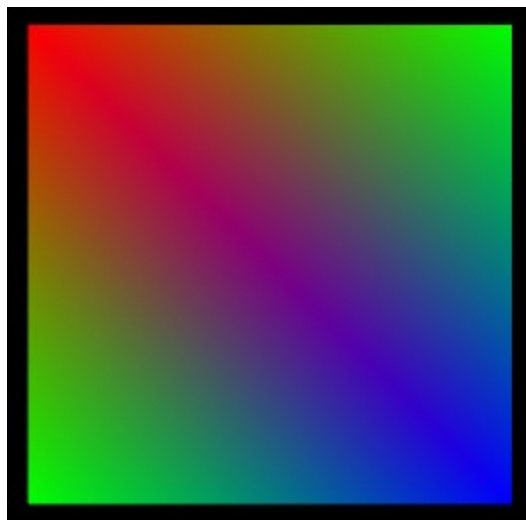
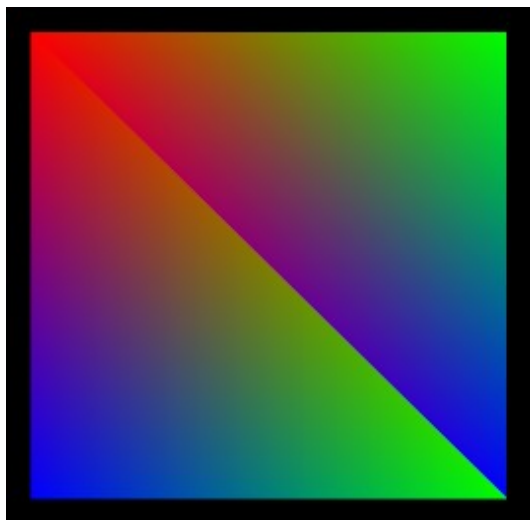
- Hem hagut de moure els polígons ja creats per posar el cercle al mig (línia 158 i 190).
- Hem jugat amb les llibreries “Threex.WindowsResize” i “Threex.FullScreen” sense èxit a la implementació.
- Per la dificultat del exercici, creiem que amb la creació d'una altre figura i amb la modificació de certs paràmetres, ja era suficient.

## Exercici 2 – Afegir color

Hem seguit amb la dinàmica del exemple\_1 i per això seguim amb el mateix codi modificant segons el tutorial el que ens demana.

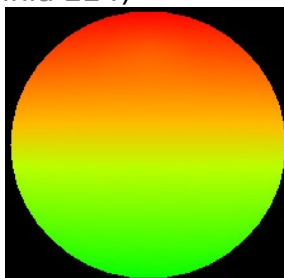
Per provar amb el color el que hem fet ha estat:

- Hem creat un fons “background” transparent (línia 53). Al no haver afegit un “background” no es pot veure la modificació per això l'hem deixat comentada.
- Hem modificat el color del quadrat intentant tenir coherència amb el color (és a dir, donant-li colors als vèrtex correctes). (línia 180)



```
squareGeometry.faces[0].vertexColors[0] = new THREE.Color(0xFF0000);  
//VERMELL  
squareGeometry.faces[0].vertexColors[1] = new THREE.Color(0x00FF00);  
//VERD  
squareGeometry.faces[0].vertexColors[2] = new THREE.Color(0x0000FF);  
//BLAU  
squareGeometry.faces[1].vertexColors[0] = new THREE.Color(0xFF0000);  
squareGeometry.faces[1].vertexColors[1] = new THREE.Color(0x0000FF);  
squareGeometry.faces[1].vertexColors[2] = new THREE.Color(0x00FF00);
```

- Al tenir el cercle, varem **cercar el mètode per intentar donar-li un color** com als polígons creats. El que hem fet (amb un poc de trampa), ha estat **crear una esfera** amb el mètode “SphereGeometry” i donar-li un tipus de material “MeshPhongMaterial”. (línia 217)
- Posteriorment hem creat una llum la qual només afectarà a la esfera, ja que només funciona amb dos tipus de material (i un es el “MeshPhongMaterial” que varem afegir a la esfera). (línia 224)



El codi emprat per la **esfera** i la **llum** es el següent:

```
var geometry = new THREE.SphereGeometry(1, 20, 20);  
//ESENCIAL - Afegir el tipus de material MeshPhongMaterial per poder  
afectar-li la llum  
var material = new THREE.MeshPhongMaterial({color: 0xddddddd});  
var cube = new THREE.Mesh(geometry, material);  
scene.add(cube);  
  
//Cream uan llum que afectarà a la esfera  
var light = new THREE.HemisphereLight(0xFF0000, 0x00FF00, 2);  
light.position.set(0,100,0);  
scene.add(light);
```

- No considerem canvis els fets en l'exemple\_1.
- Hem intentat sense èxit, crear un contorn al cercle i afegir un fons "background" en lloc de un simple color.

### Exercici 3 – Rotació

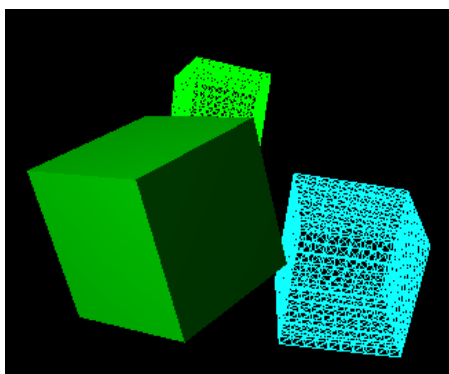
Seguint amb l'exercici 2, amb la esfera implementada. El que hem fet en aquest exercici, ha estat donar-li rotació a la esfera. Així doncs, els canvis implementats son:

- Canviar variables per "sphereMesh" per donar-li coherència (mantenint les mateixes variables que amb el "square" i el "triangle". (línia 213 i 214).
- Afegir rotació al "sphereMesh" en els eixos X i Y. (línia 241 i 242).
- Hem afegit "doubleSlide" a la esfera ja que en certes versions dona error (no era el cas, però ho varem mantenir). (línia 210)

### Exercici 4 – Formes 3D

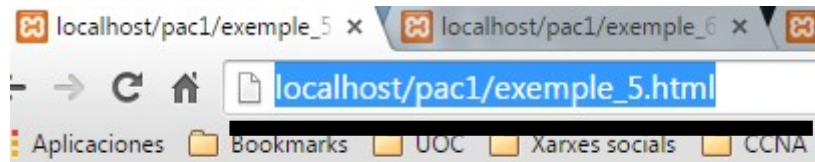
L'exercici 4, **pràcticament l'hem fet des de 0**, seguint el tutorial den Ilmari Heikkinen i el tutorial de la PAC. Pràcticament això es el que farem en [l'apartat 2](#) per a possible treball de la Practica.

- No hem creat l'esfera
- Hem creat 3 cubs un fix i la resta amb giratòria sobre aquest. La intenció era pràctica ja que per la PR es vol intentar un projecte amb plantes com a protagonistes.



- Hem afegit rotació a un objecte (línia 73) així com a rotació fixa sobre l'objecte (sobre així mateix). (línia 84)

A partir d'aquest exercici, es necessari o bé, instal·lar Python per poder veure els exercicis, o tenir instal·lat Apache (o altre servidor).. Al tenir instal·lat XAMPP, emprarem l'Apache per tal de realitzar la resta d'exercicis.



### Exercici 5 – Mapejat de textura

### Exercici 6 – Filtre de textures, il·luminació i teclat

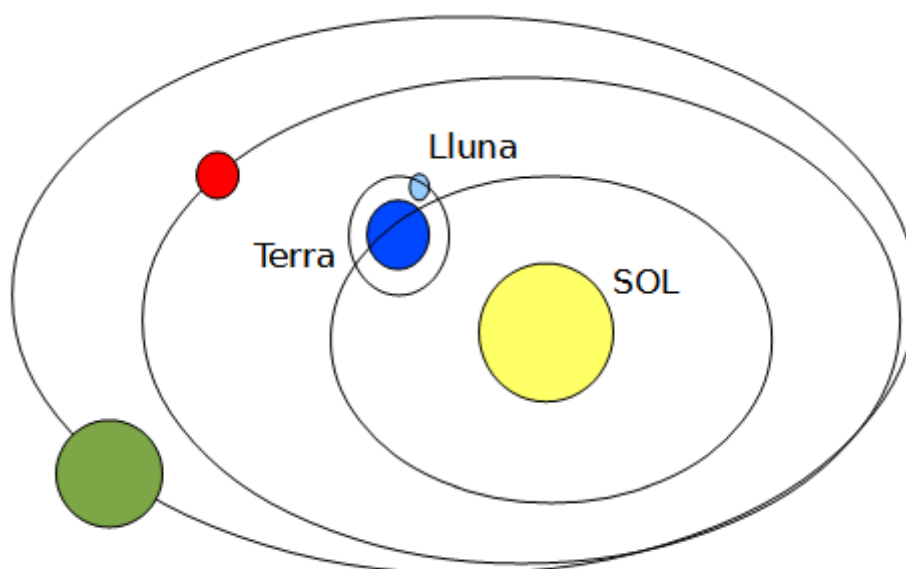
### Exercici 7 – Mescla

### Exercici 8 – Moure “bitmaps” en un espai 3D

(B) - La PAC2 consistirà en un projecte propi fet per l'alumne basat en three.js, de temàtica lliure. Es demana presentar el briefing d'aquest projecte que serà validat pel consultor.

La meua idea es intentar crear un projecte similar al **sistema solar**. Amb una esfera fixa (el sol) on la resta giraran al voltant d'aquest, més o menys propers a aquest.

- Tots els planetes giraran al voltant del sol
- La terra, tindria la lluna al seu voltant (i si es pot fer, també ho faria amb els altres planetes i els seus satèl·lits naturals).
- Afegira un text a cadascun dels planetes per poder-los diferenciar (text petit, sempre mirant cap a la càmera del projecte).
- El sol, reflectira llum. Cada planeta tindrà reflectida aquesta llum de major a menor mesura, segons la distància entre aquest i el planeta en qüestió.
- S'intentarà utilitzar una tecla per minimitzar o maximitzar la càmera.
- La càmera serà fixa des de a sobre. S'intentarà utilitzar una tecla per poder moure-la.
- S'introduirà una imatge de fons per tal de poder afegir un fons més espacial.



### Conclusió

Personalment dir que ha estat un treball que no m'ha donat temps a poder dedicar tot el temps que m'agradaria. Tot i que no pot servir d'excusa, però per motius personals no s'ha pogut dedicar tot el temps necessari. Realment ha estat la PAC que menys temps he pogut dedicar en tot el meu temps a la UOC.

Per altre banda, dir que Three.js es complicat. Primerament has de saber en la versió que estas treballant, i al anar actualitzat-se, hi ha coses que les pots trobar o desfasades, o que a la versió que s'empra actualment (la r72) s'ha de crear mitjançant un altre mètode. Aquest termes et porten a la confusió en moltíssims casos al veure que el teu codi no funciona i que ho tens com cert tutorial t'ho ha explicat.

Per finalitzar dir, que Three.js es molt potent en quant a projectes que es poden realitzar amb ell, així doncs treballarem força de cara a la Practica per intentar treure la màxima nota possible.

## Bibliografia

- HEIKKINEN, ILMARI. "Three.js basics". [en línia]. <http://fhtr.org/BasicsofThreeJS/#2>
- JOHANES, RAID. "Three.js tutorial". [en línia]. <http://www.johannes-raida.de/index.htm>
- ANGELOV, MARTI. (2013). "Three.js examples". [en línia]. <http://tutorialzine.com/2013/09/20-impressive-examples-for-learning-webgl-with-three-js/>
- Usuari: STEMKOSKI. (2013). "Three.js examples". [en línia]. <http://stemkoski.github.io/Three.js/#hello-world>
- Usuari: MRDOOB. "Three.js rar". [en línia]. <https://github.com/mrdoob/three.js>
- DIRKSEN, JOS. (2013). "All 109 Examples from my book on Three.js for Three.js version r63". [en línia]. <http://www.smartjava.org/content/all-109-examples-my-book-threejs-threejs-version-r63>
- Diversos posts o articles de "Stack Overflow":
  - LONESWAN. (2012). "Generating a regular polygon". [en línia]. <http://stackoverflow.com/questions/12656138/how-to-render-2d-shape-of-points-in-three-js>
  - SERGEI. (2013). "Clickable polygons". [en línia]. <http://stackoverflow.com/questions/18787604/creating-clickable-polygons-three-js>
  - JAMES. (2012). "Displaying background color and transparency". [en línia]. <http://stackoverflow.com/questions/12368200/displaying-background-colour-through-transparent-png-on-material>
  - VALAY. (2012). "How to rotate a cylinder to a specific point". [en línia]. <http://stackoverflow.com/questions/12746011/three-js-how-do-i-rotate-a-cylinder-around-a-specific-point>
- Pàgina oficial de "Three.js". [en línia]. <http://threejs.org/>
- DEMEUSE, JASON. (2013). "Introduction to polygonal modeling and three.js". [en línia]. <http://www.smashingmagazine.com/2013/09/introduction-to-polygonal-modeling-and-three-js/>
- YADAV, AJEET. (2014). "Creating a 3D cube". [en línia]. <http://www.awwwards.com/creating-3d-cube-a-practical-guide-to-three-js-with-live-demo.html>