**Università degli studi di Udine**

**Corso di Immagini e Multimedialità – 2018-19**

prof. Vito Roberto

**Relazione finale**

**Zanutta Riccardo**

120169

# ESERCIZIO 3.1

##### Traccia dell’esercizio

Creare una scena con un color cube.

1. **Fare le traslazioni sui singoli assi (x, y, z)**

Cosa succede quando mantenendo fisso il punto di visualizzazione e traslare:

* + Per valori positivi della x
  + Per valori negativi della x
  + Per valori positivi della y
  + Per valori negativi della y
  + Per valori positivi della z
  + Per valori negativi della z

1. **Fare le rotazioni sui singoli assi (x, y, z)**

Cosa succede quando mantenendo fisso il punto di visualizzazione e ruotare:

* + Per valori positivi della x (provare con 90°, 180°, …)
  + Per valori negativi della x
  + Per valori positivi della y
  + Per valori negativi della y
  + Per valori positivi della z
  + Per valori negativi della z

1. **Fare le scalature sui singoli assi (x, y, z)**

Cosa succede quando mantenendo fisso il punto di visualizzazione e scalare:

* + Per valori positivi della x
  + Per valori negativi della x
  + Per valori positivi della y
  + Per valori negativi della y
  + Per valori positivi della z
  + Per valori negativi della z

##### Scenegraph

BG

TG

Transform3D

Color Cube

##### Codice relativo alle traslazioni

public TransformGroup createSubGraph() {

//Creazione nuovo TransformGroup

TransformGroup transform = new TransformGroup();

//Creazione ed aggiunta del cubo

transform.addChild(new ColorCube(0.2));

//Creazione nuova trasformazione

Transform3D t3d = new Transform3D();

//TRASLAZIONE

t3d.setTranslation(new Vector3d(0.0d, 0.0d, -0.5d)); //Vector3d(x, y, z)

transform.setTransform(t3d);

return transform;

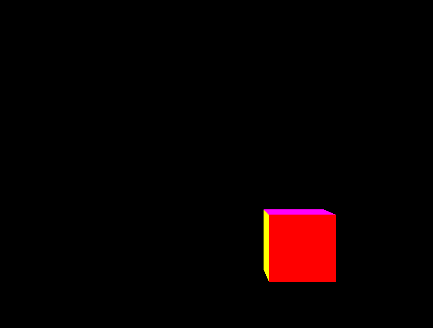
}

##### Schermate relative alle traslazioni

|  |  |  |
| --- | --- | --- |
|  | Valori positivi | valori negativi |
| X |  |  |
| Y |  |  |
| Z |  |  |

Effettuare una traslazione in modo che il cubo sia visto in basso a dx e risulti più piccolo.

t3d.setTranslation(new Vector3d(0.5d, -0.5d, -1.0d));



Per valori positiva della X l’oggetto viene spostato a destra, per valori negativi viene sposta a sinistra.

Per valori positivi della Y l’oggetto viene spostato in alto, per valori negativi viene spostato in basso.

Per valori positiva della Z l’oggetto viene avvicinato, per valori negativi viene allontanato.

##### Codice relativo alle rotazioni

public TransformGroup createSubGraph() {

//Creazione nuovo TransformGroup

TransformGroup transform = new TransformGroup();

//Creazione ed aggiunta del cubo

transform.addChild(new ColorCube(0.2));

//Creazione nuova trasformazione

Transform3D t3d = new Transform3D();

//ROTAZIONE

t3d.rotX(Math.*PI*/3); //uguale per rotY e rotZ

transform.setTransform(t3d);

return transform;

}

##### Schermate relative alle rotazioni

|  |  |  |
| --- | --- | --- |
|  | Valori positivi | valori negativi |
| X |  |  |
| Y |  |  |
| Z |  |  |

##### Codice relativo alle scalature

//Funzione che crea oggetto TG e aggiungo cubo come suo figlio

public TransformGroup createSubGraph() {

//Creazione nuovo TransformGroup

TransformGroup transform = new TransformGroup();

//Creazione ed aggiunta del cubo

transform.addChild(new ColorCube(0.2));

//Creazione nuova trasformazione

Transform3D t3d = new Transform3D();

//SCALATURA

t3d.setScale(new Vector3d(2.0d, 1.0d, 1.0d)); //x positiva

//t3d.setScale(new Vector3d(-2.0d, 1.0d, 1.0d)); //x negativa

//t3d.setScale(new Vector3d(1.0d, 2.0d, 1.0d)); //y positiva

//t3d.setScale(new Vector3d(1.0d, -2.0d, 1.0d)); //y negativa

//t3d.setScale(new Vector3d(1.0d, 1.0d, 6.0d)); //z positiva

//t3d.setScale(new Vector3d(1.0d, 1.0d, -6.0d)); //z negativa

transform.setTransform(t3d);

return transform;

}

##### Schermate relative alle scalature

|  |  |  |
| --- | --- | --- |
|  | Valorei positivi | valori negativi |
| X |  |  |
| Y |  |  |
| Z |  |  |