

© Professors d'IDI – Curs 2012-2013

Bloc_2: Transformacions Geomètriques i Models

Sessions i Objectius

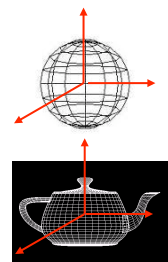
- Sessió 1 –seccions 1 a 3- : Transformacions Geomètriques
 - Objectes glut
 - Entendre el funcionament de les transformacions geomètriques per: posicionar i animar objectes.
 - Utilització en OpenGL.
 - Exercici.
- Sessió 2 –seccions 4 i 5-:
 - Utilització de les transformacions geomètriques per a inspeccionar objectes.
 - Models geomètrics (OBJ) i visualització en OpenGL.
- Sessió 3 –secció 6- :
 - Aplicació resum de conceptes: sistema solar amb astronauta.

Secció 1: Objectes glut

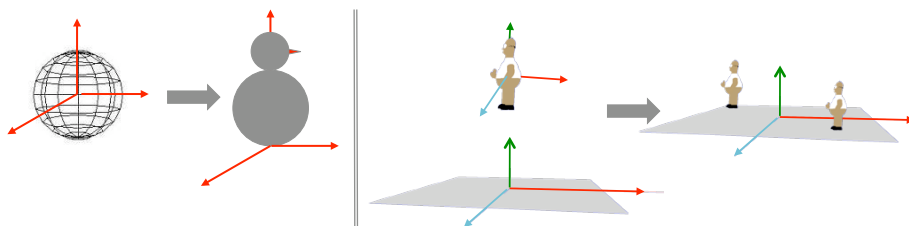
```
#include <GL/gl.h>
#include <GL/freeglut.h>
void refresh (void)
{ glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);
  glBegin(GL_TRIANGLES);
    glVertex3f(-0.5,-0.5,0.0);
    glVertex3f(0.5,0.0,0.0);
    glVertex3f(0.0,0.5,0.0);
  glEnd();
  glutSwapBuffers();
}

int main(int argc, const char *argv[])
{ glutInit(&argc, ((char **) argv);
  glutInitDisplayMode(GLUT_DEPTH | GLUT_DOUBLE | GLUT_RGBA);
  glutInitWindowSize(600,600);
  glutCreateWindow("IDI: Practiques OpenGL");
  glutDisplayFunc (refresh);
  glutMainLoop();
  return 0;
}
```

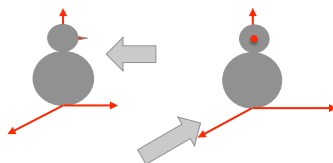
glutWireSphere(GLdouble radius, GLint slices, GLint stacks)
 glutWireCone(GLdouble base, GLdouble height, GLint slices, GLint stacks)
 glutWireTeapot(GLdouble size)
 ...



Secció 2: TG. Necessitat



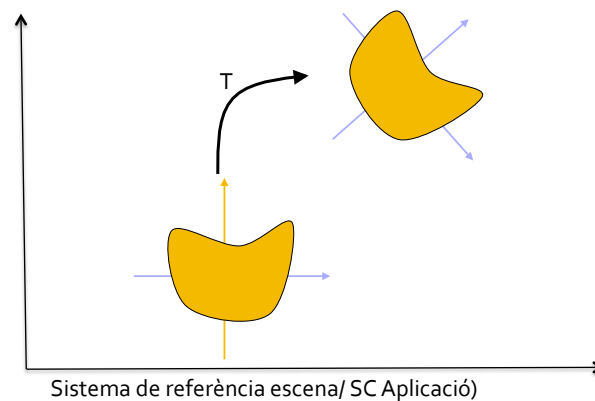
Ubicar Objectes escena, instàncies a objectes, animació d'objectes,
 ubicar objectes respecte càmera per inspecció



Necessitem modificar
 coordenades dels Vèrtexs =>
 TG (gir, escalat, translació o
 composició d'elles)

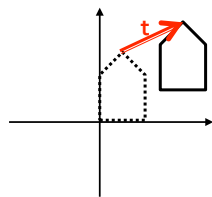
Secció 2: TG. Repàs (mireu apunts racó)

- Transformacions geomètriques bàsiques: escalat, rotacions, translació



Exemples 2D de Transformacions

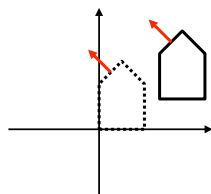
- Translació de punts



$$\begin{aligned}x' &= x + t_x \\ y' &= y + t_y\end{aligned}$$

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} x + t_x \\ y + t_y \\ 1 \end{pmatrix}$$

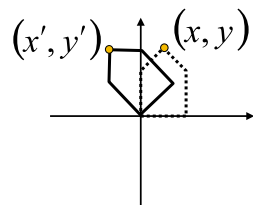
- Translacions no afecten als **vectors**



$$\begin{pmatrix} x' \\ y' \\ 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 0 \end{pmatrix} = \begin{pmatrix} x \\ y \\ 0 \end{pmatrix}$$

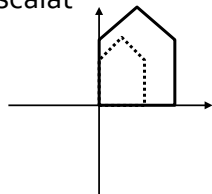
Exemples 2D (en coord, homogènies)

■ Rotacions



$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

■ Escalat



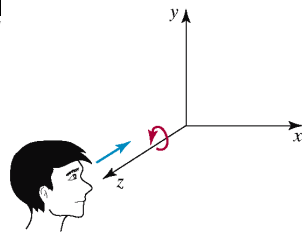
$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} s_x x \\ s_y y \\ 1 \end{pmatrix}$$

Matriu d'escalat 3D

$$S(s_x, s_y, s_z) = \begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$S(s_x, s_y, s_z) \cdot P = \begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} xs_x \\ ys_y \\ zs_z \\ 1 \end{bmatrix}$$

Matriu de rotació sobre l'eix Z

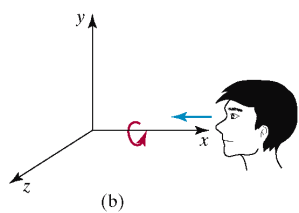


(a)

$$\begin{aligned}x' &= x \cos \alpha - y \sin \alpha \\y' &= y \sin \alpha + x \cos \alpha \\z' &= z\end{aligned}$$

$$R_Z(\alpha) \cdot P = \begin{bmatrix} \cos \alpha & -\sin \alpha & 0 & 0 \\ \sin \alpha & \cos \alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x \cos \alpha - y \sin \alpha \\ x \sin \alpha + y \cos \alpha \\ z \\ 1 \end{bmatrix}$$

Matriu de rotació sobre l'eix X

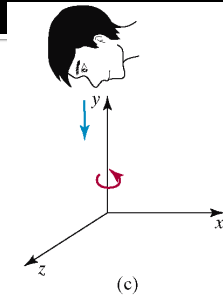


(b)

$$R_X(\alpha) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha & 0 \\ 0 & \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$R_X(\alpha) \cdot P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha & 0 \\ 0 & \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x \\ y \cos \alpha - z \sin \alpha \\ y \sin \alpha + z \cos \alpha \\ 1 \end{bmatrix}$$

Matriu de rotació sobre l'eix Y



$$R_y(\alpha) = \begin{bmatrix} \cos \alpha & 0 & \sin \alpha & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \alpha & 0 & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$R_y(\alpha) \cdot P = \begin{bmatrix} \cos \alpha & 0 & \sin \alpha & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \alpha & 0 & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x \cos \alpha + z \sin \alpha \\ y \\ -x \sin \alpha + z \cos \alpha \\ 1 \end{bmatrix}$$

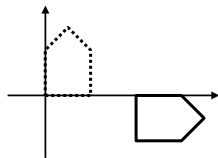
Matriu de translació

$$T(t_x, t_y, t_z) = \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T(t_x, t_y, t_z) \cdot P = \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x + t_x \\ y + t_y \\ z + t_z \\ 1 \end{bmatrix}$$

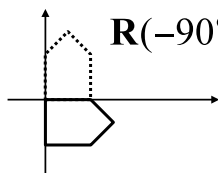
Composició de Transformacions

- Imaginem que volem

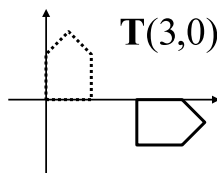


No es pot fer amb cap de les matrius anteriors

- Cal composar/efectuar dues transformacions (moviments)



$R(-90^\circ)$



$T(3,0)$

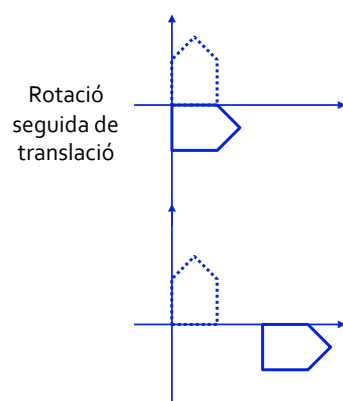
$$M = T(3,0) \cdot R(-90^\circ)$$

$$P' = T(3,0) \cdot (R(-90^\circ) P) = (T(3,0) \cdot R(-90^\circ)) P = M \cdot P$$

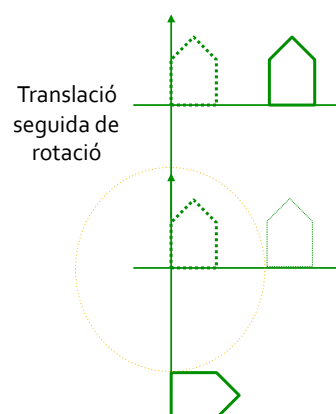
Composició de Transformacions

$$T(3,0) \cdot R(-90^\circ) \neq R(-90^\circ) \cdot T(3,0)$$

- Multiplicació de matrius no és commutativa



Rotació
seguida de
translació



Translació
seguida de
rotació

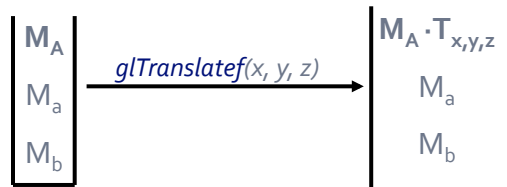
Transformations geomètriques i OpenGL

- OpenGL manté *dues piles de matrius* (la MODELVIEW i la PROJECTION). Només una pot estar activa per a ser modificada. Per defecte: la pila MODELVIEW

- La matriu activa M_A és sempre la del top de la *pila activa*

- Te instruccions per a crear matrius de TG que afecten a la M_A :

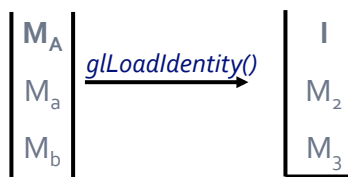
- `void glTranslate{fd}(TYPE x, TYPE y, TYPE z);`
- `void glRotate{fd}(TYPE angle, TYPE x, TYPE y, TYPE z);`
- `void glScale{fd}(TYPE x, TYPE y, TYPE z);`



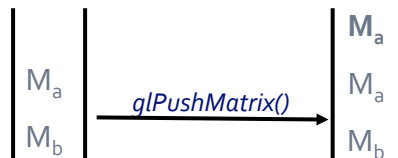
$M_A = M_A * TG$
Noteu l'ordre de la multiplicació!!!

Transformations geomètriques i OpenGL

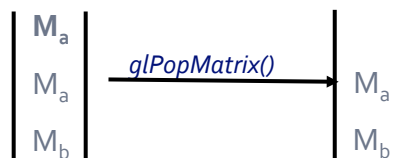
- Substituir la matriu al top de la pila: (`glLoadIdentity`, `glLoadMatrix*`)



- Duplicar la matriu del top de la pila: `glPushMatrix()`



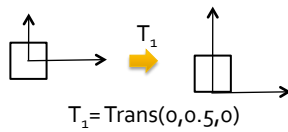
- Eliminar la matriu del top de la pila: `glPopMatrix()`



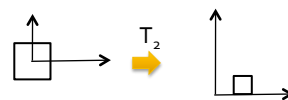
Transformations Geomètriques i OpenGL

- Quan s'envia a pintar un vèrtex V , li aplica primer la matriu del top de la pila MODELVIEW (el "mou")

- $V_transformat = MODELVIEW_TOP * V$



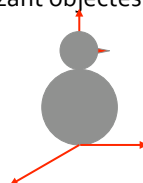
```
glMatrixMode(GL_MODELVIEW);
glTranslatef(0, 0.5, 0);
glutWireCube(1);
```



```
glMatrixMode(GL_MODELVIEW);
glTranslatef(0.75, 0.25, 0);
glScalef(0.5, 0.5, 0.5);
glutWireCube(1);
```

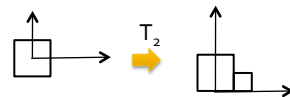
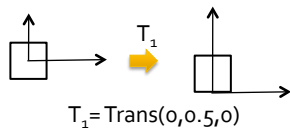
Què heu de fer en 1ra sessió Bloc 2?

- Pintar algun objecte glut (secció 1)
- Utilitzar OpenGL per aplicar TG a un objecte (secció 2)
 - Entendre els paràmetres de les crides i composició d'operacions
 - Recordeu que aplica la matriu del top de la pila
 - Utilitzeu callbacks de teclat i ratolí per modificar TG
 - FEU LES PROVES QUE US INDIQUEM
- Utilitzar OpenGL per a aplicar diferents TG als diferents objectes de l'escena
 - Caldrà Push/Pop Matrius
- Crear una escena utilitzant objectes glut (secció 3)



Model transformations i OpenGL

- Quan s'envia a pintar un vèrtex V li aplica primer la matriu del top de la pila MODELVIEW (el "mou")
 - $V_transformat = MODELVIEW_{TOP} * V$



```
glMatrixMode(GL_MODELVIEW);
glPushMatrix();
glTranslatef(0, 0.5, 0);
glutWireCube(1);
glPopMatrix();
glPushMatrix();
glTranslatef(0.75, 0.25, 0);
glScalef(0.5, 0.5, 0.5);
glutWireCube(1);
glPopMatrix();
```