Outline

1. The Coordinate Axes
2. Homogeneous Coordinates
3. Geometric Transformations
   - Translations
   - Rotations
   - Scalings
   - Reflections
4. Assignment
There are three mutually orthogonal axes: the $x$-axis, the $y$-axis, and the $z$-axis.

In the standard viewing position, the $x$- and $y$-axes look the same as in the usual 2D coordinate system.

The positive $z$-axis points towards the viewer; the negative $z$-axis points away.
The Coordinate Axes
The Coordinate Axes
The Coordinate Axes
Points and Homogeneous Coordinates

- Points are represented in homogeneous coordinates \((x, y, z, w)\), where \(w \neq 0\).
- The standard 3-dimensional coordinates are given by \((x/w, y/w, z/w)\).
- For this reason, we usually want \(w = 1\).
- However, some transformations may change the \(w\)-coordinate.
Definition (Isometry)

An *isometry* is a geometric transformation that preserves distances and angles.

- **Isometries**
  - Translations
  - Rotations
  - Reflections

- **Non-isometries**
  - Scalings
  - Shears
All of these transformations may be performed in 2 or 3 dimensions.

When they are performed in 2 dimensions, typically the $z$-coordinate of the points is 0.

They are handled the same way whether in 2 or 3 dimensions.
A translation is a displacement in a particular direction.

- A translation is defined by specifying the displacements $\Delta x$, $\Delta y$, and $\Delta z$.

$$
\begin{align*}
x' &= x + \Delta x \\
y' &= y + \Delta y \\
z' &= z + \Delta z
\end{align*}
$$
Translations
Translations

$$\Delta x = 8, \Delta y = 1, \Delta z = 0$$
Translations in OpenGL

- The function
  
  \[ \text{glTranslate}*(\Delta x, \Delta y, \Delta z) \]

  performs a translation through the displacement \( \Delta x, \Delta y, \Delta z \).

- To translate an object, this function should be called before the object is drawn.
Translation Example

Example (Translation)

```c
glTranslatef(4.0, 2.0, -3.0);
glBegin(GL_TRIANGLES);
    glVertex3f(0.0, 0.0, 0.0);
    glVertex3f(1.0, 0.0, 0.0);
    glVertex3f(0.0, 1.0, 0.0);
glEnd();
```
Translation Example

Example (Translations)

- The code.
- The executable.
Rotations

Definition (Rotation)

A rotation turns about a point \((a, b)\) through an angle \(\theta\).

- Generally we rotate about the origin.
- Using the \(z\)-axis as the axis of rotation, the equations are

\[
\begin{align*}
x' &= x \cos \theta - y \sin \theta \\
y' &= x \sin \theta + y \cos \theta \\
z' &= z
\end{align*}
\]
Rotations
Rotations

$z$-axis, $\theta = 60^\circ$
Rotations in OpenGL

The function

$$\text{glRotate}^\ast (\theta, \ a, \ b, \ c)$$

will rotate about the line \(x = at, \ y = bt, \ z = ct\) through an angle \(\theta\), measured in degrees.

Typically, we rotate about
- The \(x\)-axis \((1, 0, 0)\),
- The \(y\)-axis \((0, 1, 0)\), or
- The \(z\)-axis \((0, 0, 1)\).
Geometric Transformations

Robb T. Koether

The Coordinate Axes
Homogeneous Coordinates
Geometric Transformations
Translations
Rotations
Scalings
Reflections
Assignment

Rotation Example

Example (Rotation)

```c
glRotatef(90.0, 0.0, 0.0, 1.0);
glBegin(GL_TRIANGLES);
    glVertex3f(0.0, 0.0, 0.0);
    glVertex3f(1.0, 0.0, 0.0);
    glVertex3f(0.0, 1.0, 0.0);
glEnd();
```
Direction of Rotation

- The direction of rotation is determined by the “right-hand rule.”
- If you point your thumb in the positive direction of the axis of rotation, then when you curl your fingers, they will curl in the positive direction of rotation.
- This rule works even if you are left-handed, as long as you use your right hand.
Rotation Example

Example (Rotations)

- The code.
- The executable.
Scaling

Definition (Scaling)

A scaling is an expansion or contraction in the $x$, $y$, and $z$ directions by scale factors $s_x$, $s_y$, and $s_z$, and centered at a point $(a, b, c)$.

- Generally we center the scaling at the origin.

$$\begin{align*}
x' &= s_x x \\
y' &= s_y y \\
z' &= s_z z
\end{align*}$$
Scaling
Scaling

\[ s_x = 3, \quad s_y = 2, \quad s_z = 1 \]
Scalings in OpenGL

- The function
  
  \[ \text{glScale}(s_x, s_y, s_z) \]

  will scale the drawing by factors \( s_x \), \( s_y \), and \( s_z \) in the \( x \)-, \( y \)-, and \( z \)-directions.

- The center of the scaling is the origin.

- Never use a scale factor of 0.
Scaling Example

Example (Scaling)

```c
glScalef(4.0, 2.0, 1.0);
glBegin(GL_TRIANGLES);
    glVertex3f(0.0, 0.0, 0.0);
    glVertex3f(1.0, 0.0, 0.0);
    glVertex3f(0.0, 1.0, 0.0);
glEnd();
```
Example (Scalings)

- The code.
- The executable.
Reflections

Definition (Reflection)
A reflection is a reversal of an object with respect to a line in 2 dimensions or a plane in 3 dimensions.

- Generally we reflect in a line or plane through the origin.
Reflections
Reflection in the $x$-Axis
Reflection in the $y$-Axis
Reflection in the Line $y = -x$
The function

$$\text{glScalef}(sx, sy, sz)$$

will perform a reflection if two of the factors are 1 and one is $-1$.

For example, to reflect in the $y$-axis, use

$$\text{glScalef}(1.0, -1.0, 1.0);$$
Reflection Example

Example (Reflection)

```c
GLfloat(1.0, -1.0, 1.0);
glBegin(GL_TRIANGLES);
  glVertex3f(0.0, 0.0, 0.0);
  glVertex3f(1.0, 0.0, 0.0);
  glVertex3f(0.0, 1.0, 0.0);
glEnd();
```
Reflection Example

Example (Reflections)

- The code.
- The executable.
Reflections

- If we reflect in two axes at once (or in sequence), that amounts to a rotation of $180^\circ$ in the axis perpendicular to those two axes.
- For example, `glScalef(-1.0, -1.0, 1.0);` is equivalent to `glRotatef(180.0, 0.0, 0.0, 1.0);`
To accomplish a reflection in an arbitrary line \( L: ax + by = 0 \) through the origin,

First, reflect in the x-axis (\( y = 0 \)).

Then rotate through twice the angle between the line \( L \) and the x-axis.
Reflections

To accomplish a reflection in an arbitrary plane $P : ax + by + cz = 0$ through the origin,

- First, reflect in the $xy$-plane ($z = 0$).
- Then rotate through twice the angle between the plane $P$ and the $xy$-plane and about the line of intersection between the plane $P$ and the $xy$-plane.
Homework

- Read Section 4.3 – transformations.
- Read Section 4.6 – affine transformations.
- Read Section 4.7 – translations, rotations, and scalings.