Advanced Game Engine Creation

3D Transformations

Lecture 4

Objectives

So far we have:

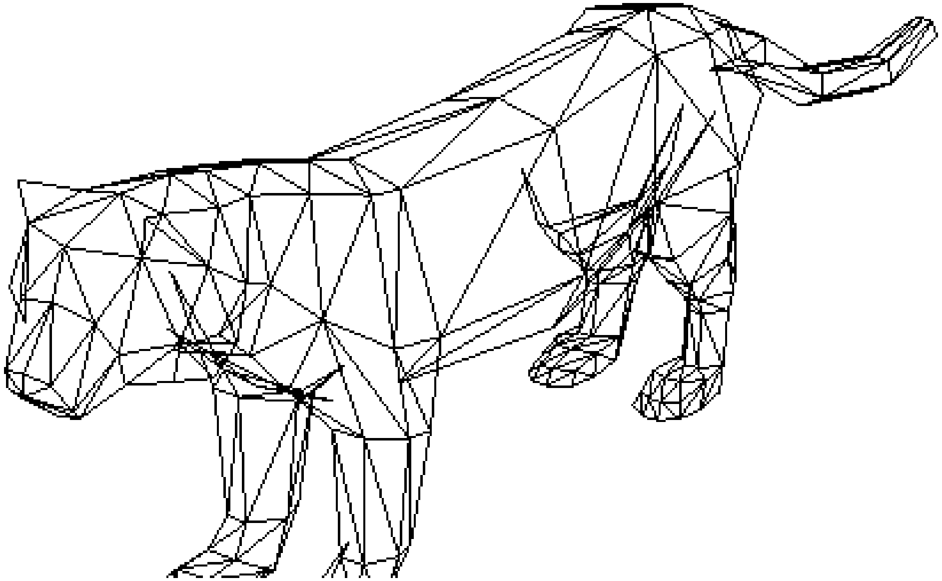
- Looked at 3D representation of points and vectors
- Introduced matrices and how to multiply them
- Drawn and manipulated OpenGL primitives

In this lecture, we will

- Discuss how 3D transformations can be represented using matrices
- introduce homogeneous transformation matrices
- Apply matrix multiplication to transform objects in 3D space

Recall

- game world is represented in 3D space by models made up of triangles made up of vertices



Geometric transformations

- remember we keep only one copy of each model
  - Ideally centred on the origin
- a copy of the model is scaled, rotated and placed into position every frame
  - multiple copies for multiple objects
- scaling about the origin
  - multiply the coordinates of each vertex by a scale factor
- rotation about an axis
  - use trigonometry to calculate the new position of each vertex
- translation
  - add a translation vector to each coordinate

What are Translations?

- Translations move every point by the same amount
  - eg move every point 5 units ‘up’
    - this is a ***translation*** of 5 units parallel to the y-axis
  - eg move every point 10 units to the right
    - this is a ***translation*** of 10 units parallel to the x-axis

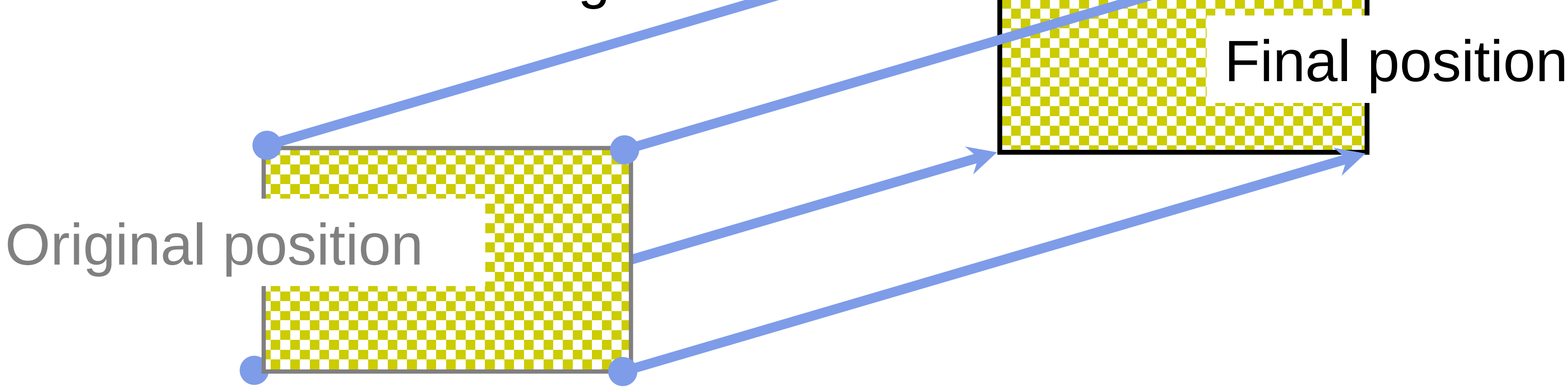
a vector (10, 5) could be added to each point to move it 10 units to the right and 5 units up

What are Translations?

- ***Translations*** will move every point in a fixed direction given by a ***vector*** (t_x , t_y, t_z)
- we can translate a point (x_i, y_i, z_i) to a new position (x_f, y_f, z_f) by adding a vector (t_x , t_y, t_z)

$$(x_i, y_i, z_i) + (t_x , t_y, t_z) = (x_f, y_f, z_f)$$

Translation of a rectangle



The Identity Matrix

$$I = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

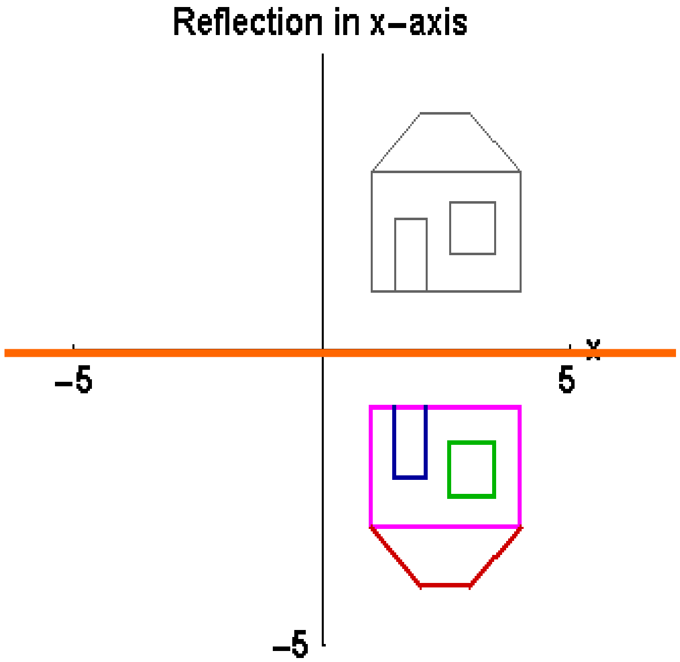
- multiplying any point(s) by this matrix leaves it unchanged (fixed)

The Identity Matrix

- recall from Tutorial 2:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 4 & 2 & 2 \\ 3 & 2 & -2 & 8 \\ 0 & 9 & 1 & 1 \\ 1 & -3 & 6 & 2 \end{bmatrix} = \begin{bmatrix} 1 & 4 & 2 & 2 \\ 3 & 2 & -2 & 8 \\ 0 & 9 & 1 & 1 \\ 1 & -3 & 6 & 2 \end{bmatrix}$$

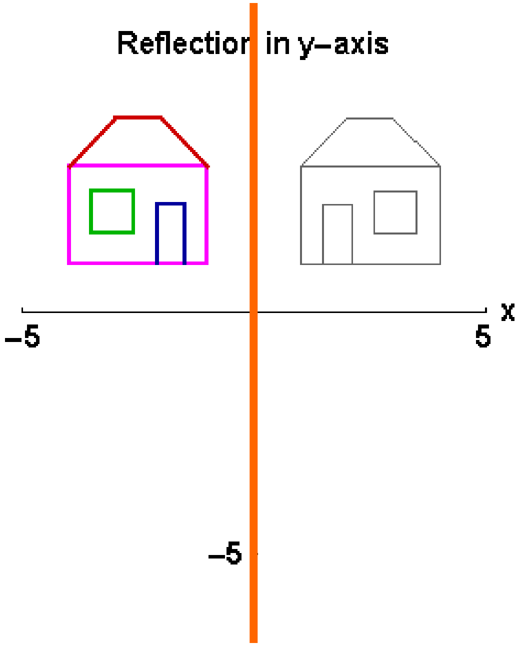
Reflection in x-axis



$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

- multiplying any point by the above matrix will
  - keep the x-coordinate the same
  - reverse the sign of the y-coordinate
- as if the shape is reflected in a mirror positioned along the x-axis
- try it!

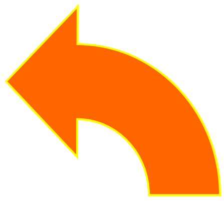
Reflection in y axis



$$\begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Rotations - Conventions

- In Computing and Mathematics the notation is to take anticlockwise rotations as Positive
- So a rotation of 90^0 means an anticlockwise rotation of 90^0



Rotation of θ degrees about the z axis

- general rotation matrix about the z axis

$$\begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Rotate 90^0 about the z-axis

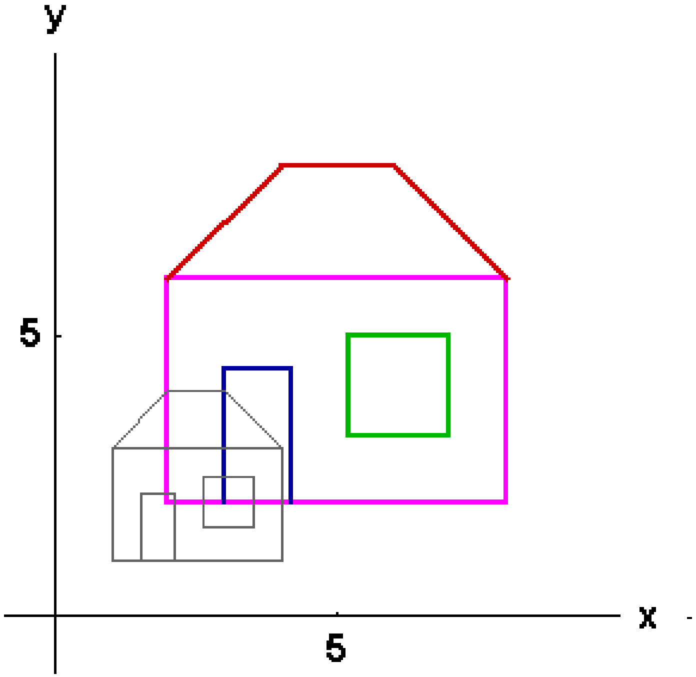
- $\cos(90) = 0 \qquad \sin(90) = 1$

$$\begin{bmatrix} \cos(90) & -\sin(90) & 0 \\ \sin(90) & \cos(90) & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 2 \\ 5 \\ 1 \end{bmatrix} = \begin{bmatrix} -5 \\ 2 \\ 1 \end{bmatrix}$$

Enlargement or Scaling , Scale Factor s

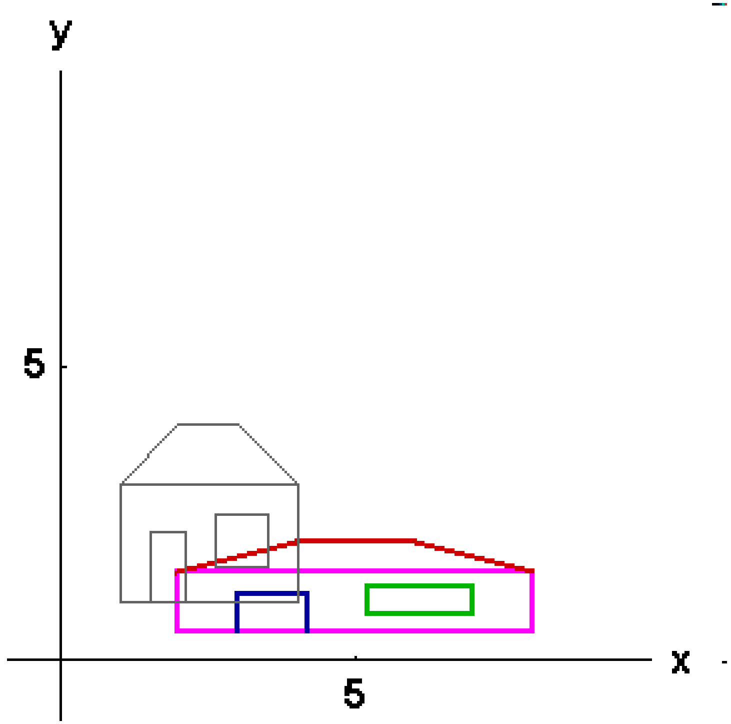
- An enlargement or scaling
  - centre origin, scale factor s:



$$\begin{bmatrix} s & 0 & 0 \\ 0 & s & 0 \\ 0 & 0 & s \end{bmatrix}$$

Stretches, Scale Factors s_x and s_y

- A stretch of scale factor s_x parallel to the x axis, s_y parallel to the y axis and s_z parallel to the z axis is



$$\begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & s_z \end{bmatrix}$$

Scaling examples

- uniform scaling:

$$\begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ 3 \\ 0 \end{bmatrix} = \begin{bmatrix} 2 \\ 6 \\ 0 \end{bmatrix}$$

- non-uniform scaling (stretch)

$$\begin{bmatrix} 2 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 3 \\ 0 \end{bmatrix} = \begin{bmatrix} 2 \\ 15 \\ 0 \end{bmatrix}$$

Problem

- we can translate the vertices in a model by adding the translation vector to each vertex
- we can rotate, scale or stretch the model by multiplying each vertex by the appropriate transformation matrix
- it would be nice to represent all transformations by matrices
- which we could multiply together to get a compound transformation matrix
  - and then apply it to each vertex in the model
- solution – use homogeneous coordinates and matrices

Homogeneous coordinates for 3D Geometry

- Definition:
- The homogeneous coordinates of a point P in 3D space are given by

$$(x , y , z, w) \text{ where } w \neq 0$$

- This corresponds to the 3D (X,Y,Z) coordinate system point:

$$(x / w , y / w , z / w)$$

Standardised Homogeneous coordinates for 3D Geometry

- For this course we will only use **Standardised** homogeneous coordinates
- These are homogeneous coordinates with **w = 1**
- So $(\mathbf{x}, \mathbf{y}, \mathbf{z}, \mathbf{1}) \leftrightarrow (\mathbf{x}, \mathbf{y}, \mathbf{z})$
- homogeneous $\leftrightarrow$ **3D**

Standardised Homogeneous coordinates for 3D Geometry

- In column vector format:

$$\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \Longleftrightarrow \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

Standardised Homogeneous

Non-homogeneous

Homogeneous 4x4 Matrices

- Next consider homogeneous 4x4 matrices
- They represent geometric transformations called **Affine Transformations**
  - Scalings and rotations about any point
  - Reflections and shears with respect to any lines
  - Translations

General Template for a 4x4 Homogeneous Matrix

$$\left[\begin{array}{ccc|c} a & b & c & t_x \\ c & d & e & t_y \\ f & g & h & t_z \\ \hline 0 & 0 & 0 & 1 \end{array}\right]$$

Note:
The matrix is partitioned into four blocks

General Template for a 4x4 Homogeneous Matrix

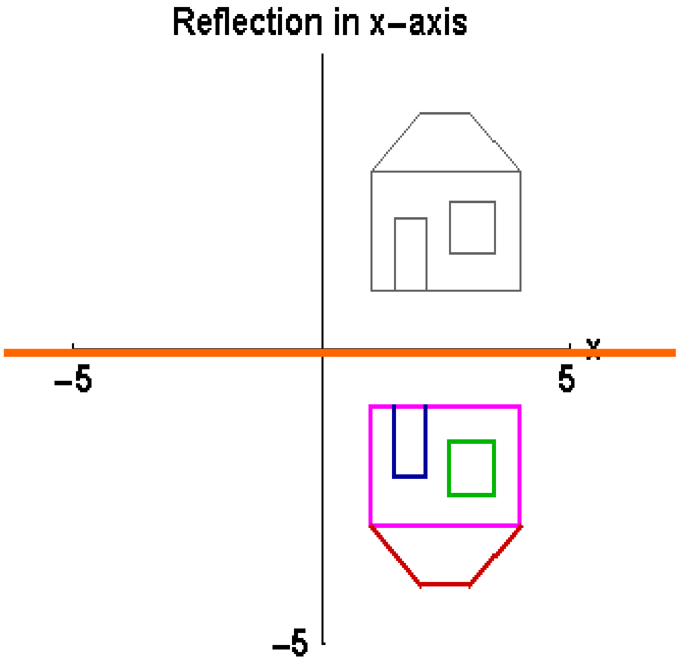
- The 3x3 matrix $\mathbf{M_{3x3}}$ represents a reflection, rotation, or enlargement centred on the origin
  - its entries are identical to the 3 x 3 matrices
- In these cases t_x t_y and t_z are all set equal to ZERO

General Template for a 4x4 Homogeneous Matrix

- To represent a **translation** of (t_x , t_y , t_z)
  - set M_{3x3} equal to the 3x3 identity matrix
  - set t_x t_y and t_z to the required values

- *Sample Templates follow*

Homogeneous matrix template: Reflection in the x-axis



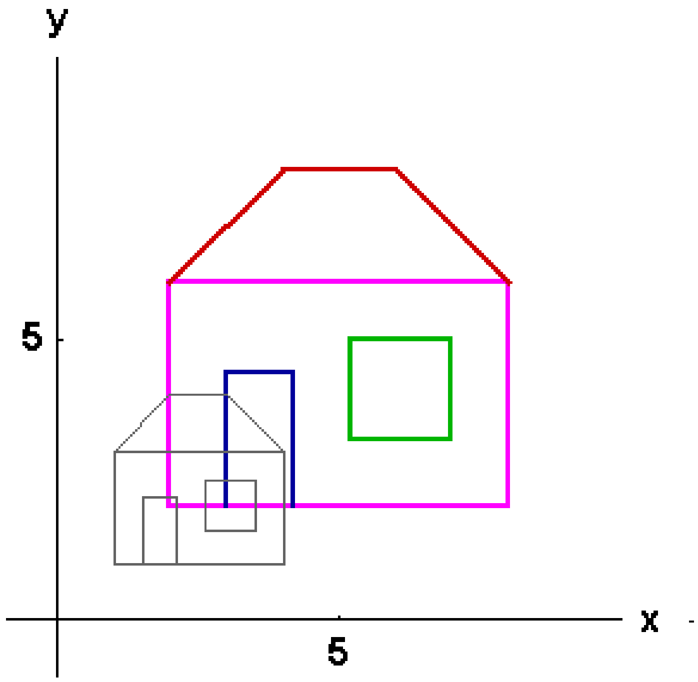
$$\left[\begin{array}{ccc|c} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 0 & 1 \end{array}\right]$$

Homogeneous matrix template: Rotation of θ degrees about the z-axis

$$\begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Homogeneous matrix template: Enlargement , Scale Factor s

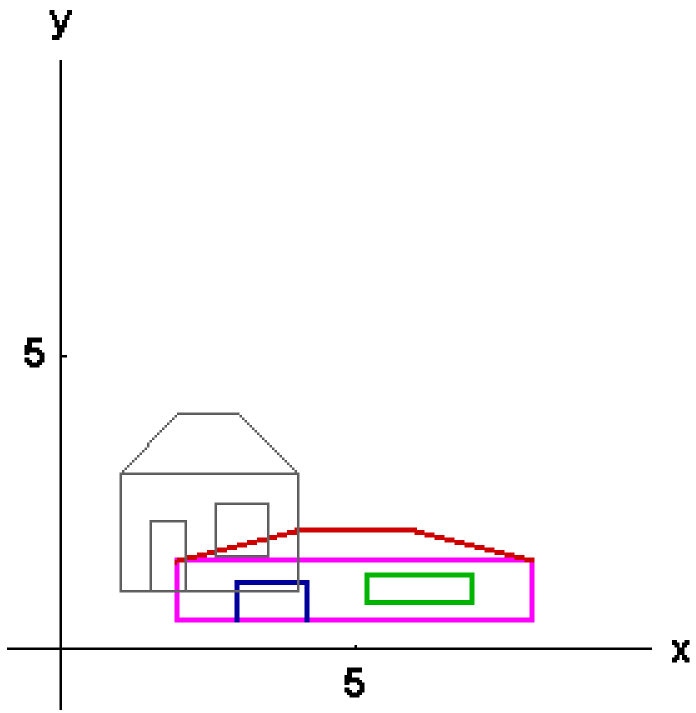
- An enlargement centre Origin, scale factor s is:



$$\left[\begin{array}{ccc|c} s & 0 & 0 & 0 \\ 0 & s & 0 & 0 \\ 0 & 0 & s & 0 \\ \hline 0 & 0 & 0 & 1 \end{array}\right]$$

Homogeneous matrix template: Stretches, Scale Factors s_x s_y and s_z

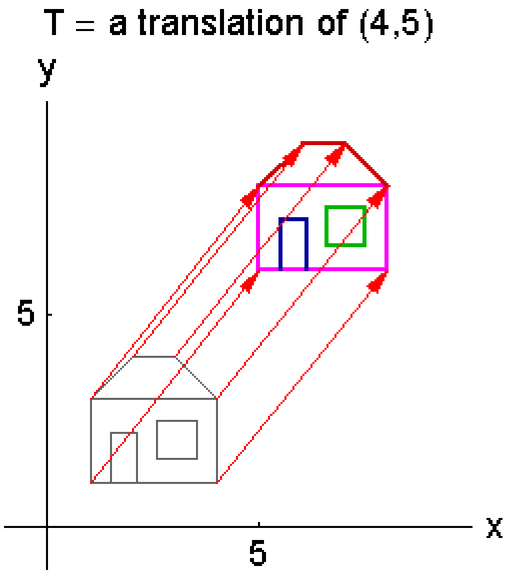
- A stretch of scale factor s_x parallel to the x axis, s_y parallel to the y axis and s_z parallel to the z axis is



$$\left[\begin{array}{ccc|c} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ \hline 0 & 0 & 0 & 1 \end{array}\right]$$

Homogeneous matrix template:
A translation of (t_x, t_y, t_z)

- The matrix is



$$\left[\begin{array}{ccc|c} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ \hline 0 & 0 & 0 & 1 \end{array}\right]$$

Forming Complex Transformations

by concatenating matrix products

Homogeneous matrix template: General Cases

- The previous slides gave the ‘common’ cases
- More complex transformations are built up by combining transformations
  - this involves ***matrix multiplications*** ***(****see appendix****)***
  - the order of the matrices is important

Homogeneous matrix template: General Cases

- **Example**: R = a rotation of 60^0 about the point (15,20)
- Solution:
  - we know the matrix, ***R60***, for a rotation of 60^0 about the *origin (0,0)*
  - we know the matrix, ***T***, for a translation from (15,20) to the origin (0,0)
  - we know the matrix, T^{-1}, for a translation from the origin (0,0) back to (15,20)

Homogeneous matrix template: General Cases

- **Example**: R = a rotation of 60^0 about the point (15,20)
- Solution:
  - *using the three matrices on the previous slide*

$$R = T^{-1} \bullet R60 \bullet T$$

Homogeneous matrix template: General Cases

- Example: E = enlargement scale factor 2, centre (-6,7)
- Solution:
  - we know the matrix, ***E2**,* for an enlargement scale factor 2 centre the *origin (0,0)*
  - we know the matrix, ***T***, for a translation from (-6,7) to the origin (0,0)
  - we know the matrix, T^{-1}, for a translation from the origin (0,0) back to (-6,7)

Homogeneous matrix template: General Cases

- Example: E = an enlargement scale factor 2, centre (-6,7)
- Solution:
  - *using the three matrices on the previous slide*

$$\mathbf{R} = \mathbf{T}^{-1} \bullet \mathbf{E2} \bullet \mathbf{T}$$

Appendix: Matrix Multiplication

- ***To Form entry i , j in the Matrix Product* A.B**
- **1. Multiply** corresponding elements in ***row i*** by corresponding elements in ***column j*** (ie first element in ***row i*** by first element in ***column j*** , ...)
- **2. Sum** all the results from stage 1
- **3.** Put the sum from stage 2 into entry ***i , j***

Appendix: Matrix Multiplication - 5
Worked Example: The Entry in row2, column 3

$$\begin{bmatrix} 1 & -2 & 1 & 3 \\ 0 & 1 & -1 & 1 \\ 4 & -3 & 4 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 3 & 1 & -1 & 0 \\ 1 & 1 & -1 & 1 \\ -2 & 2 & 1 & -1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} \square & \square & \square & \square \\ \square & \square & -2 & \square \\ \square & \square & \square & \square \\ \square & \square & \square & \square \end{bmatrix}$$

row 2 by column 3

(return to main sequence)

Summary

Today we have

- Discussed how 3D transformations can be represented using matrices
- introduced homogeneous transformation matrices
- Applied matrix multiplication to transform objects in 3D space

Practical

- Apply matrix multiplication to transform objects in 3D space
- perform transformations in OpenGL

Further reading

http://www.opengl-tutorial.org/beginners-tutorials/tutorial-3-matrices/

http://matrix.reshish.com/multiplication.php