

# Transformations



## Topics

---

**Purpose of transformations**

**Gallery of transformations: rotations, translation, ...**

**Composing transformations**

**Representing transformations as matrices**

**Hierarchical transformations**

# Transformations

---

**What?**

$$\mathbf{x}' = \mathbf{T}(\mathbf{x})$$

**Why?**

**Modeling**

- Create objects in convenient coordinates
- Multiple instances of a prototype shape
- Kinematics of linkages/skeletons - robots

**Viewing**

- Windows and device independence
- Virtual camera: parallel/perspective projections

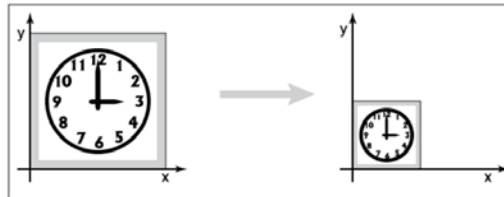
```
glTranslate(1,1,0);  
glRotate(45.,0,0,1);  
  
square();
```

## Types of Transformations

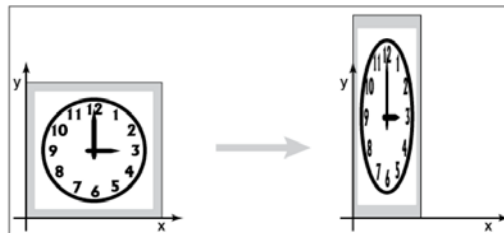
## Scale

---

Uniform



Nonuniform

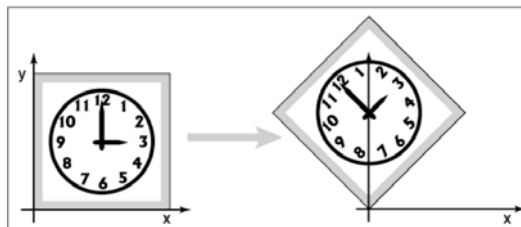


CS148 Lecture 4

Pat Hanrahan, Winter 2007

## Rotate

---

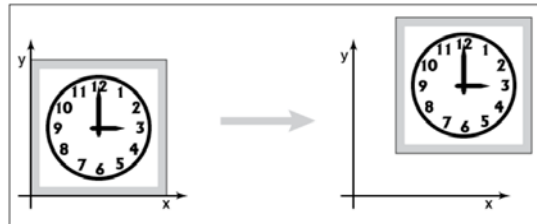


CS148 Lecture 4

Pat Hanrahan, Winter 2007

## Translate

---

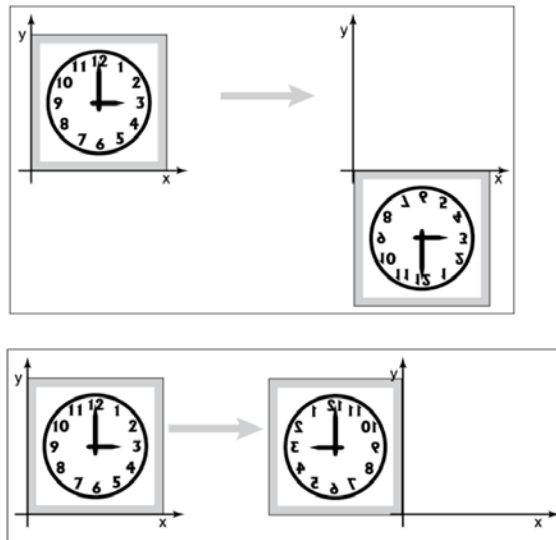


CS148 Lecture 4

Pat Hanrahan, Winter 2007

## Reflect

---

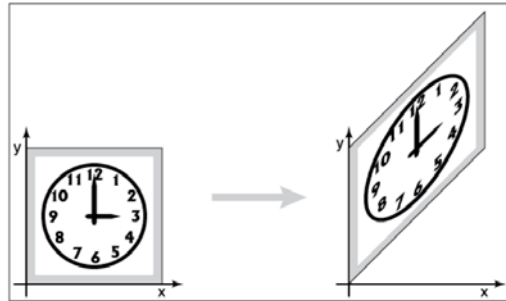
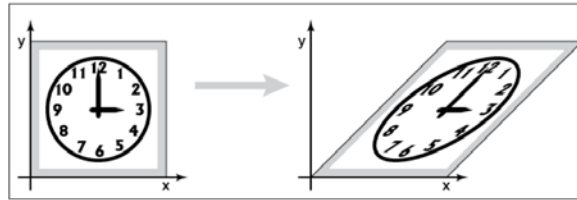


CS148 Lecture 4

Pat Hanrahan, Winter 2007

## Shear

---



CS148 Lecture 4

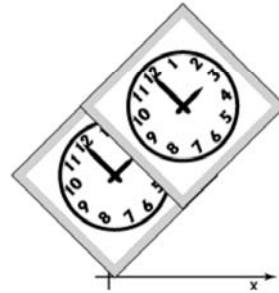
Pat Hanrahan, Winter 2007

## Composing Transformations

## Combining Translation and Rotation



$R(45)$

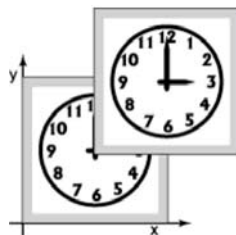


$T(1,1) R(45)$

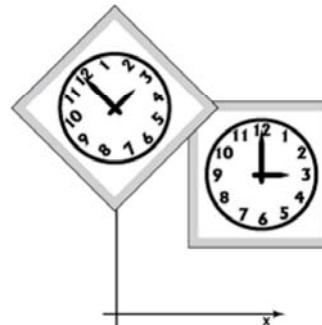
CS148 Lecture 4

Pat Hanrahan, Winter 2007

## Combining Translation and Rotation



$T(1,1)$

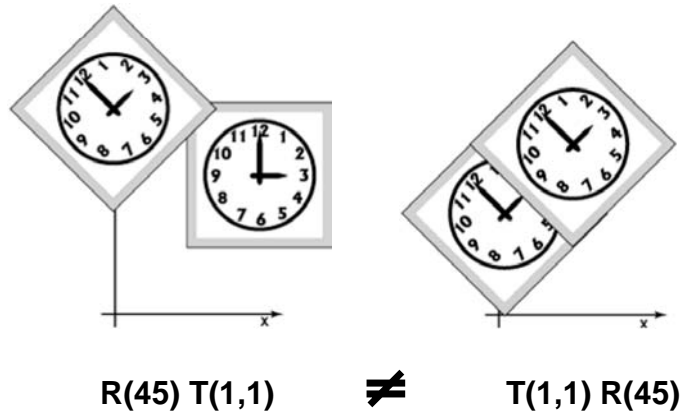


$R(45) T(1,1)$

CS148 Lecture 4

Pat Hanrahan, Winter 2007

## Order Matters

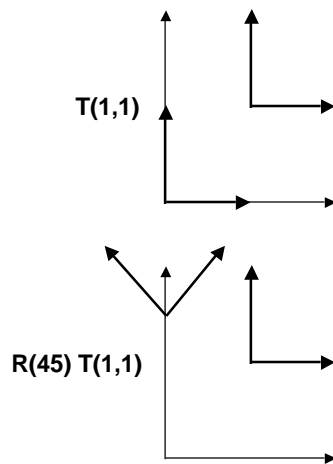


CS148 Lecture 4

Pat Hanrahan, Winter 2007

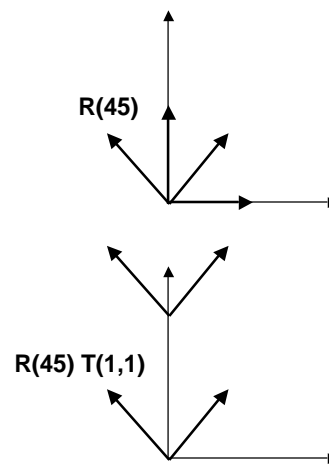
## Two Interpretations

Applied to an Object  
Read right to left



CS148 Lecture 4

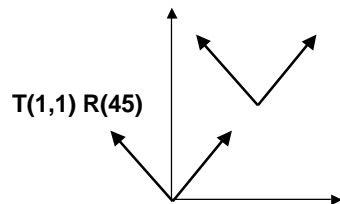
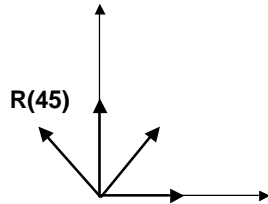
Applied to a Coordinate System  
Read left to right



Pat Hanrahan, Winter 2007

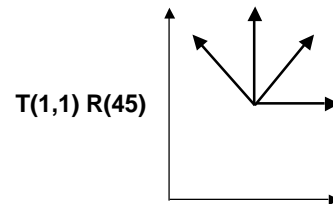
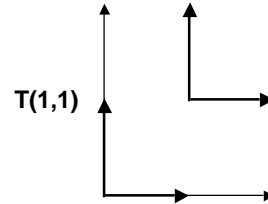
## Two Interpretations

Applied to an Object  
Read right to left



CS148 Lecture 4

Applied to a Coordinate System  
Read left to right



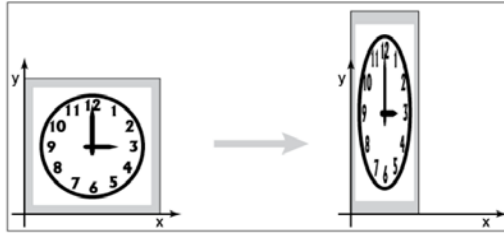
Pat Hanrahan, Winter 2007

## Math of Transformations (Matrices)



## Scale

---



$$x' = s_x x$$

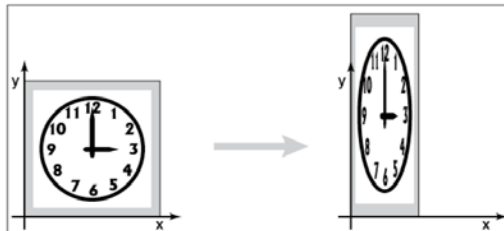
$$y' = s_y y$$

CS148 Lecture 4

Pat Hanrahan, Winter 2007

## Scale

---



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

CS148 Lecture 4

Pat Hanrahan, Winter 2007

## Linear Transformations = Matrices

---

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} m_{xx} & m_{xy} \\ m_{yx} & m_{yy} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

$$\mathbf{x}' = \mathbf{M} \mathbf{x}$$

$$x' = m_{xx} x + m_{xy} y$$

$$y' = m_{yx} x + m_{yy} y$$

CS148 Lecture 4

Pat Hanrahan, Winter 2007

## Why Called Linear Transformations?

---

Because lines are transformed into lines

Equation of a line:

$$\mathbf{x} = (1 - \alpha) \mathbf{p} + \alpha \mathbf{q}$$

$$\begin{aligned} \mathbf{x}' = \mathbf{M} \mathbf{x} &= (1 - \alpha) \mathbf{M} \mathbf{p} + \alpha \mathbf{M} \mathbf{q} \\ &= (1 - \alpha) \mathbf{p}' + \alpha \mathbf{q}' \end{aligned}$$

$$\mathbf{p}' = \mathbf{M} \mathbf{p}$$

$$\mathbf{q}' = \mathbf{M} \mathbf{q}$$

CS148 Lecture 4

Pat Hanrahan, Winter 2007

## Coordinate Frames

---

$$\begin{bmatrix} m_{xx} \\ m_{yx} \end{bmatrix} = \begin{bmatrix} m_{xx} & m_{xy} \\ m_{yx} & m_{yy} \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} m_{xy} \\ m_{yy} \end{bmatrix} = \begin{bmatrix} m_{xx} & m_{xy} \\ m_{yx} & m_{yy} \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

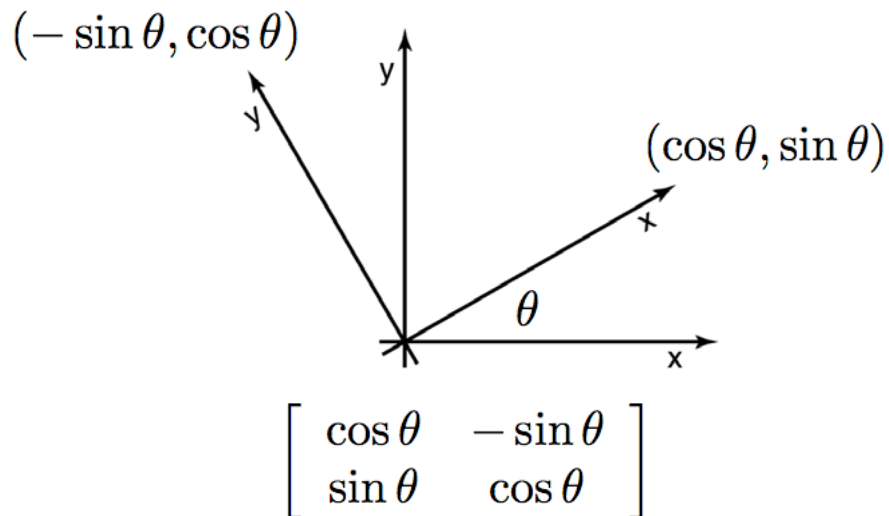
Can interpret columns of the matrix as the positions of the new x and y axes

CS148 Lecture 4

Pat Hanrahan, Winter 2007

## Rotation Matrix

---

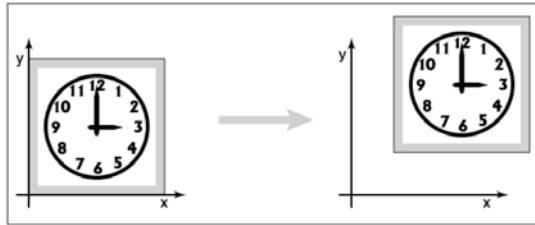


CS148 Lecture 4

Pat Hanrahan, Winter 2007

## Translations

---



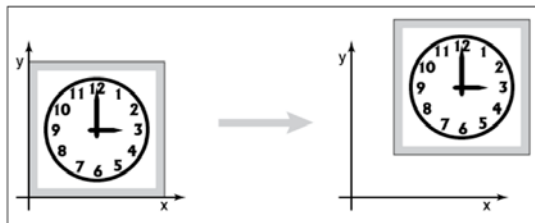
$$\begin{aligned}x' &= x + t_x \\ y' &= y + t_y\end{aligned}$$

CS148 Lecture 4

Pat Hanrahan, Winter 2007

## Translations

---



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

CS148 Lecture 4

Pat Hanrahan, Winter 2007

# Hierarchical Transformations

## Current Transformation (Matrix)

---

- Graphics systems maintain a *current transformation matrix (CTM)*. All geometry is transformed by the CTM.
- The CTM defines the current or *object* or *local* coordinate system. All geometry is defined in the current coordinate system.
- Transformation commands are concatenated onto the ctm. Note: The last transformation specified is the first to be performed.

$$CTM = CTM * T$$

- Transformations may be pushed and popped from a stack. This makes it easy to model complex assemblies of parts (robot arms, etc.)

## Advantages of the Matrix Formulation

- Avoid computing terms in the transformation matrix

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

This would require reevaluating sines and cosines

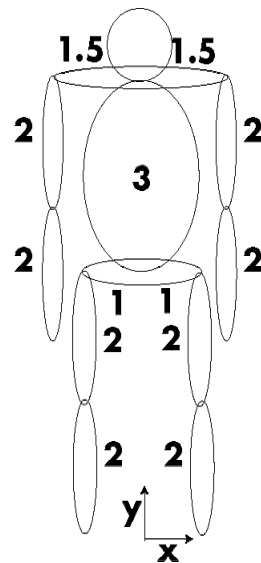
- Sequence of transforms = single transform

$$\begin{aligned}\mathbf{x}' &= (\mathbf{A}(\mathbf{B}(\mathbf{C}(\mathbf{D}\mathbf{x})))) \\&= (\mathbf{ABCD})\mathbf{x} \\&= \mathbf{M}\mathbf{x}\end{aligned}$$

CS148 Lecture 4

Pat Hanrahan, Winter 2007

## Skeleton

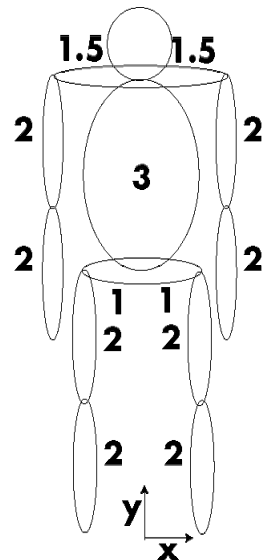


```
body
torso
head
shoulder
  larm
    upperarm
    lowerarm
    hand
  rarm
    upperarm
    lowerarm
    hand
hips
  lleg
    upperleg
    lowerleg
    foot
  rleg
    upperleg
    lowerleg
    foot
```

CS148 Lecture 4

Pat Hanrahan, Winter 2007

## Skeleton



CS148 Lecture 4

```
translate(0,4,0)
torso();
pushmatrix();
  translate(0,3,0);
  shoulder();
  pushmatrix();
    rotatey(necky);
    rotatex(neckx);
    head();
  popmatrix();
pushmatrix();
  translate(1.5,0,0);
  rotatex(lshoulderx);
  upperarm();
  pushmatrix();
    translate(0,-2,0);
    rotatex(l elbowx);
    lowerarm();
    ...
  popmatrix();
popmatrix();
...
```

Pat Hanrahan, Winter 2007