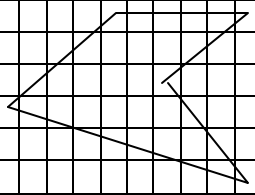
Computer Graphics

Z Sweedyk
Lecture 5
9/13/00

Outline

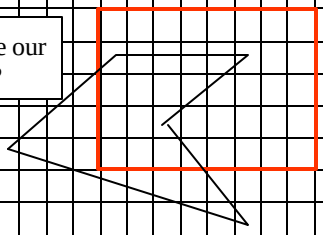
- Clipping Polygons
- Assignment 2
- 2D Transformations - intro

The World

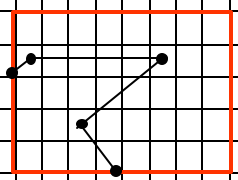


The World/The Viewport

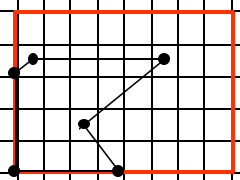
Why not use our line clipper?



Is this a polygon?



Our objective

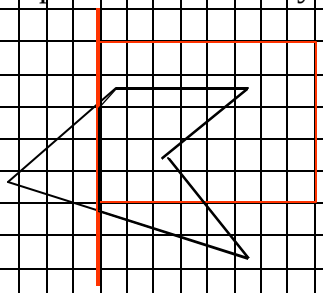


Strategy
(Sutherland-Hodgman)

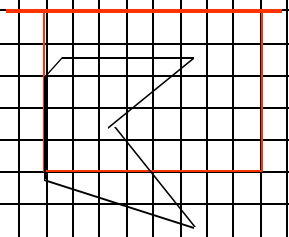
For each viewport boundary B

Clip polygon P to create polygon entirely within B

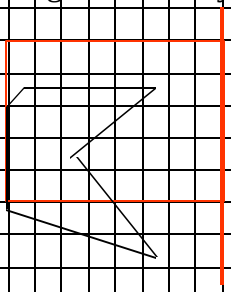
Example – Left Boundary



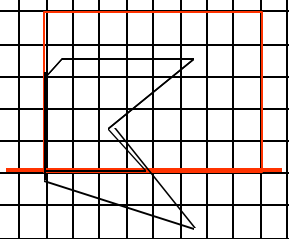
Example – Top Boundary



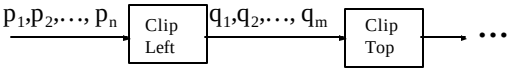
Example – Right Boundary

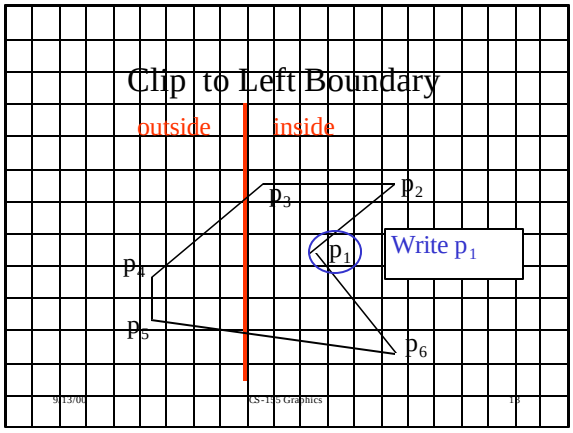


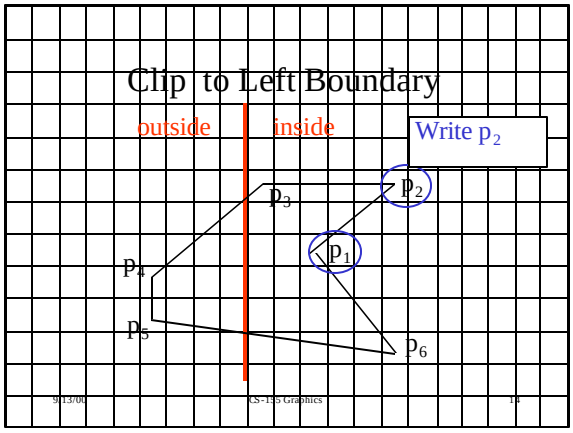
Example – Bottom Boundary

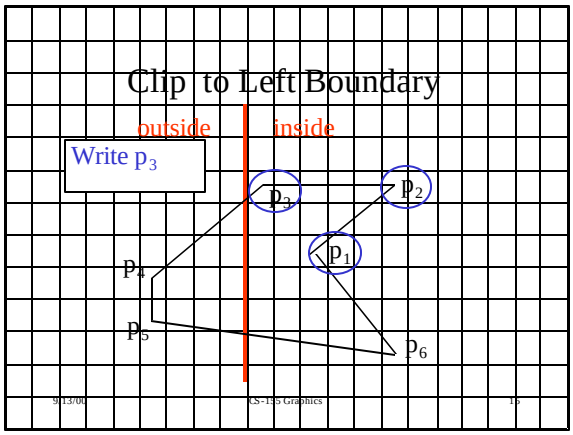


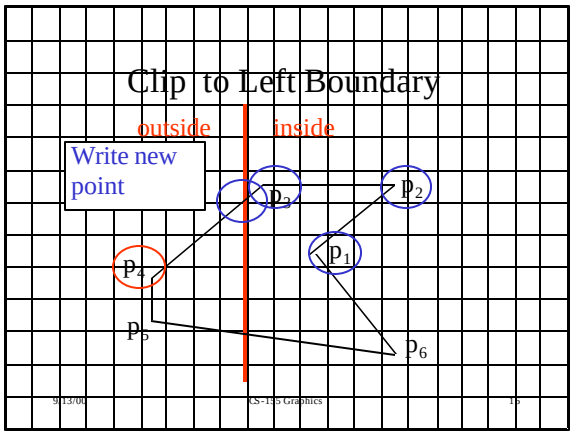
Pipeline

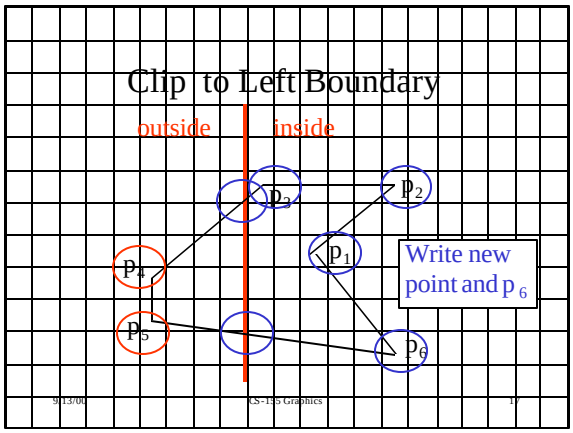












Four Cases

- p inside → q inside
- p inside → q outside
- p outside → q inside
- p outside → q outside

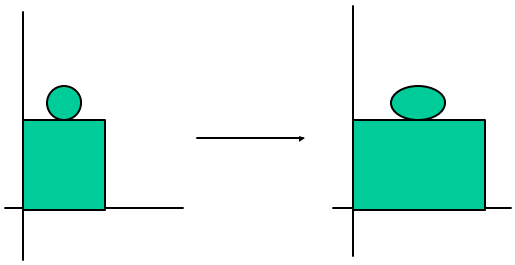
Assignment 2:

- Scan convert polygons (boundaries): 10 points
- Scan convert filled polygons using ET/AET algorithm: 60 points
- Clip polygons: 30

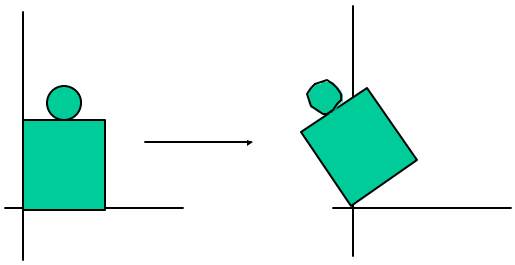
2D Transformations

- Translate
- Rotate
- Scale

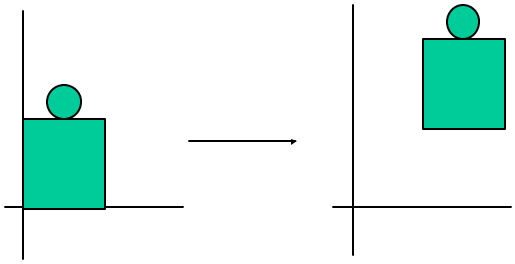
Scale



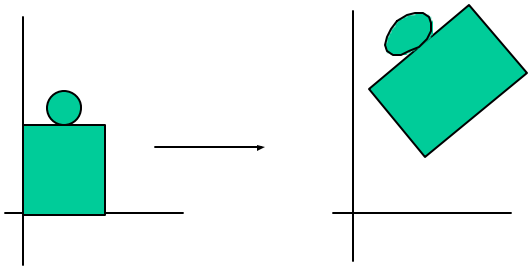
Rotate



Translate



Scale, Rotate, Translate

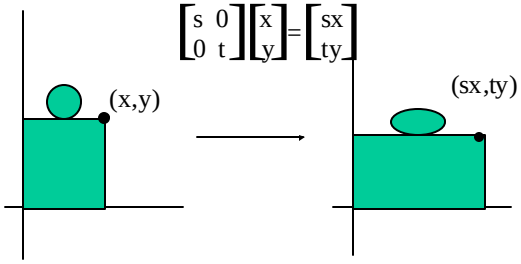


OpenGL Pipeline

Primitives defined by vertices → Vertex Transformations → Positioned Primitives defined by vertices

Scale

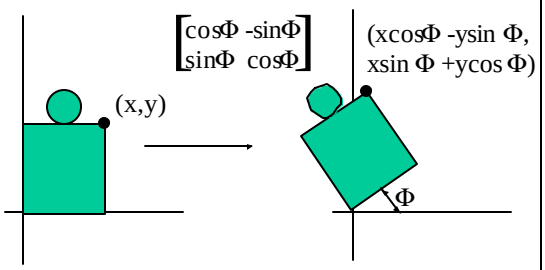
$$\begin{bmatrix} s & 0 \\ 0 & t \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix}=\begin{bmatrix} sx \\ ty \end{bmatrix}$$



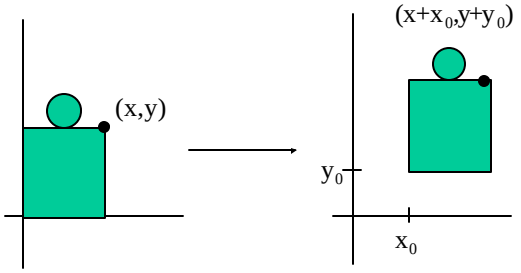
Rotate

$$\begin{bmatrix} \cos\Phi & -\sin\Phi \\ \sin\Phi & \cos\Phi \end{bmatrix}$$

$(x\cos\Phi - y\sin\Phi,\ x\sin\Phi + y\cos\Phi)$



Translate



Translate

Translation is not a linear transformation

Translate(a+b) ≠ Translate(a)+Translate(b)

OpenGL Pipeline

Primitives defined by vertices → Vertex Transformations → Positioned Primitives defined by vertices

Modeling Matrix → Vertex Transformations

Homogenous coordinates