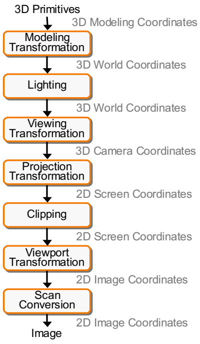
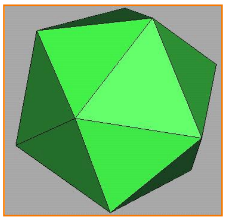
Clipping

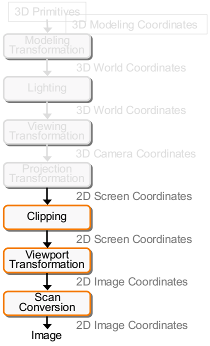
Lee Markosian
(covering for Adam Finkelstein)
Princeton University
C0S 426, Fall 2001

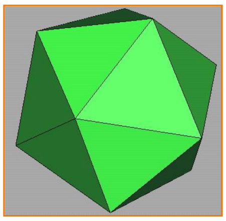
3D Rendering Pipeline (for direct illumination)


3D Primitives
3D Modeling Coordinates
Modeling Transformation
3D World Coordinates
Lighting
3D World Coordinates
Viewing Transformation
3D Camera Coordinates
Projection Transformation
2D Screen Coordinates
Clipping
2D Screen Coordinates
Viewport Transformation
2D Image Coordinates
Scan Conversion
2D Image Coordinates
Image

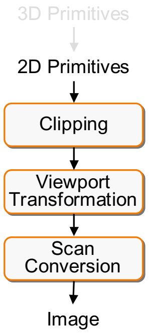


3D Rendering Pipeline (for direct illumination)


3D Primitives
3D Modeling Coordinates
Modeling Transformation
3D World Coordinates
Lighting
3D World Coordinates
Viewing Transformation
3D Camera Coordinates
Projection Transformation
2D Screen Coordinates
Clipping
2D Screen Coordinates
Viewport Transformation
2D Image Coordinates
Scan Conversion
2D Image Coordinates
Image



2D Rendering Pipeline


3D Primitives

2D Primitives

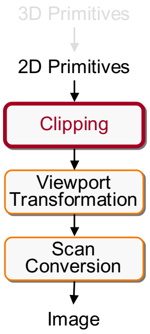
Clipping — Clip portions of geometric primitives residing outside the window

Viewport Transformation — Transform the clipped primitives from screen to image coordinates

Scan Conversion — Fill pixels representing primitives in screen coordinates

Image

2D Rendering Pipeline


3D Primitives

2D Primitives

Clipping — Clip portions of geometric primitives residing outside the window

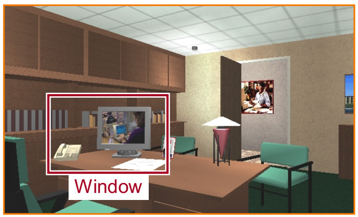
Viewport Transformation — Transform the clipped primitives from screen to image coordinates

Scan Conversion — Fill pixels representing primitives in screen coordinates

Image

Clipping

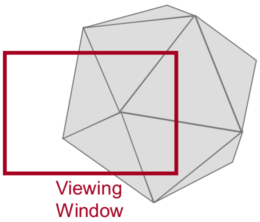
- Avoid drawing parts of primitives outside window
  - Window defines part of scene being viewed
  - Must draw geometric primitives only inside window



Screen Coordinates

Clipping

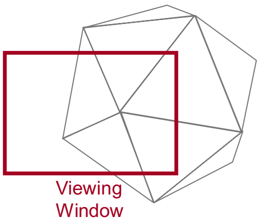
- Avoid drawing parts of primitives outside window
  - Window defines part of scene being viewed
  - Must draw geometric primitives only inside window



Viewing Window

Clipping

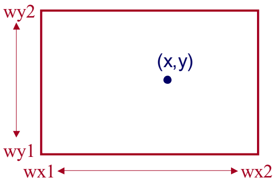
- Avoid drawing parts of primitives outside window
  - Points
  - Lines
  - Polygons
  - Circles
  - etc.



Viewing Window

Point Clipping

- Is point (x,y) inside the clip window?

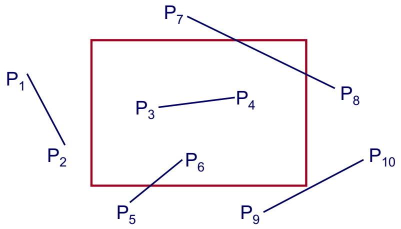

wy2
(x,y)
wy1
wx1
wx2

Window

```
inside =
   (x >= wx1) &&
   (x <= wx2) &&
   (y >= wy1) &&
   (y <= wy2);
```

Line Clipping

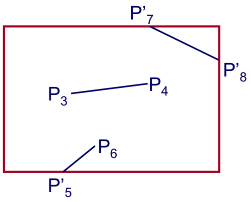
- Find the part of a line inside the clip window


P_7
P_1
P_4
P_8
P_3
P_2
P_6
P_{10}
P_5
P_9

Before Clipping

Line Clipping

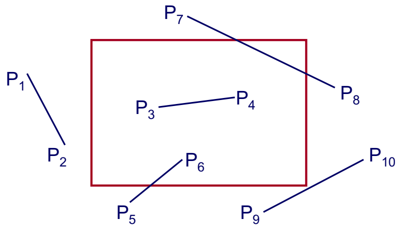
- Find the part of a line inside the clip window


P'_7
P'_8
P_3
P_4
P_6
P'_5

After Clipping

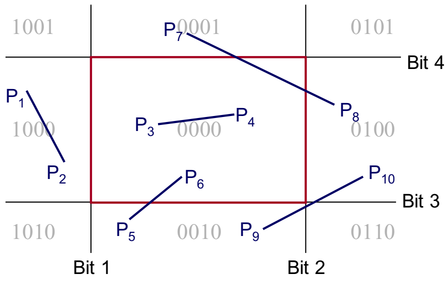
Cohen Sutherland Line Clipping

- Use simple tests to classify easy cases first


P_7
P_1
P_4
P_8
P_3
P_2
P_6
P_{10}
P_5
P_9

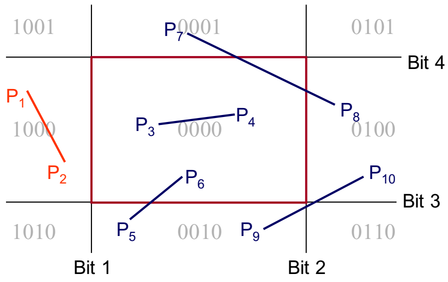
Cohen Sutherland Line Clipping

- Classify some lines quickly by AND of bit codes representing regions of two endpoints (must be 0)


1001
P_7
0001
0101
Bit 4
P_1
1000
P_2
P_3
0000
P_4
P_8
0100
P_6
P_{10}
Bit 3
1010
P_5
0010
P_9
0110
Bit 1
Bit 2

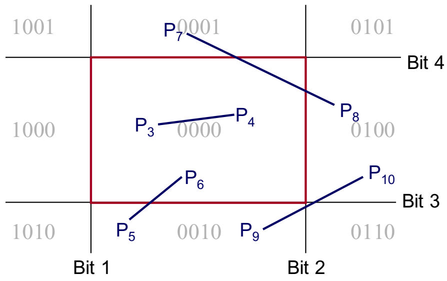
Cohen Sutherland Line Clipping

- Classify some lines quickly by AND of bit codes representing regions of two endpoints (must be 0)


1001
P_7
0001
0101
Bit 4
P_1
1000
P_2
P_3
0000
P_4
P_8
0100
P_6
P_{10}
Bit 3
1010
P_5
0010
P_9
0110
Bit 1
Bit 2

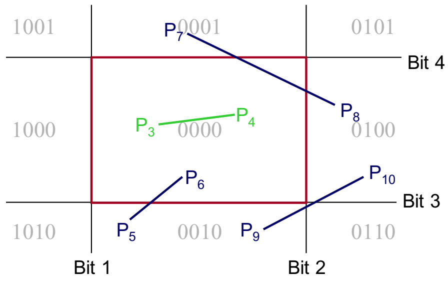
Cohen Sutherland Line Clipping

- Classify some lines quickly by AND of bit codes representing regions of two endpoints (must be 0)


1001
P_7
0001
0101
Bit 4
1000
P_3
0000
P_4
P_8
0100
P_6
P_{10}
Bit 3
1010
P_5
0010
P_9
0110
Bit 1
Bit 2

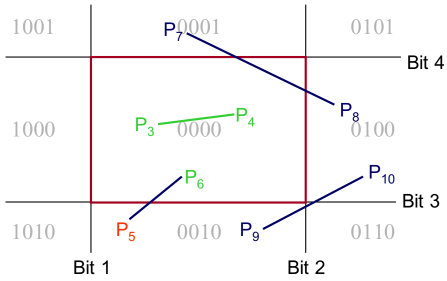
Cohen Sutherland Line Clipping

- Classify some lines quickly by AND of bit codes representing regions of two endpoints (must be 0)


1001
P_7
0001
0101
Bit 4
1000
P_3
0000
P_4
P_8
0100
P_6
P_{10}
Bit 3
1010
P_5
0010
P_9
0110
Bit 1
Bit 2

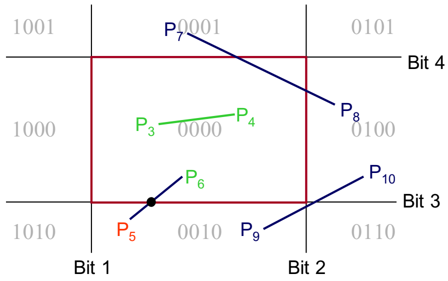
Cohen-Sutherland Line Clipping

- Compute interesections with window boundary for lines that can’t be classified quickly


1001
P_7
0001
0101
Bit 4
1000
P_3
0000
P_4
P_8
0100
P_6
P_{10}
Bit 3
1010
P_5
0010
P_9
0110
Bit 1
Bit 2

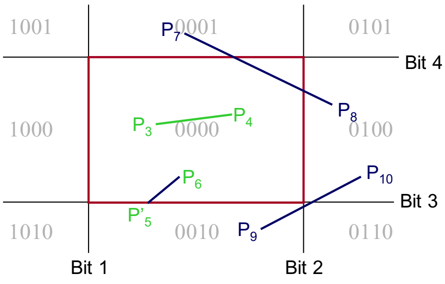
Cohen-Sutherland Line Clipping

- Compute interesections with window boundary for lines that can’t be classified quickly


1001
P_7
0001
0101
Bit 4
1000
P_3
0000
P_4
P_8
0100
P_6
P_{10}
Bit 3
1010
P_5
0010
P_9
0110
Bit 1
Bit 2

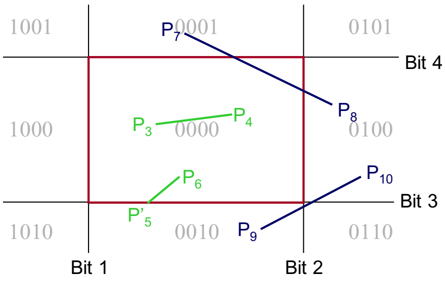
Cohen-Sutherland Line Clipping

- Compute interesections with window boundary for lines that can’t be classified quickly


1001
P_7
0001
0101
Bit 4
P_3
P_4
P_8
1000
0000
0100
P_6
P_{10}
Bit 3
P'_5
1010
0010
P_9
0110
Bit 1
Bit 2

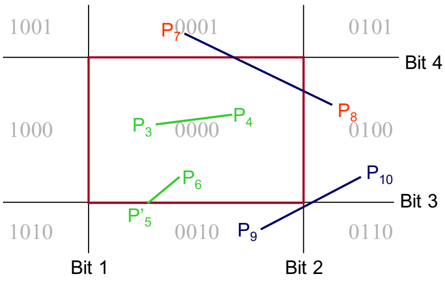
Cohen-Sutherland Line Clipping

- Compute interesections with window boundary for lines that can’t be classified quickly


1001
P_7
0001
0101
Bit 4
P_3
P_4
P_8
1000
0000
0100
P_6
P_{10}
Bit 3
P'_5
1010
0010
P_9
0110
Bit 1
Bit 2

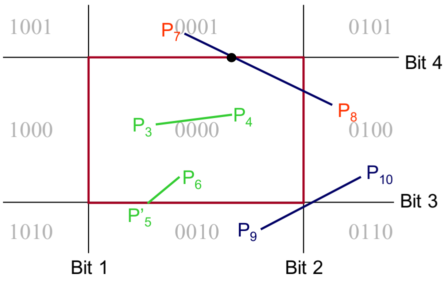
Cohen-Sutherland Line Clipping

- Compute interesections with window boundary for lines that can’t be classified quickly


1001
P_7
0001
0101
Bit 4
P_3
P_4
P_8
1000
0000
0100
P_6
P_{10}
Bit 3
P'_5
1010
0010
P_9
0110
Bit 1
Bit 2

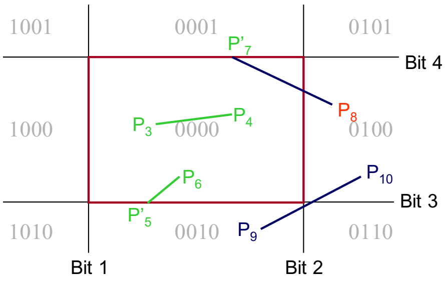
Cohen-Sutherland Line Clipping

- Compute interesections with window boundary for lines that can’t be classified quickly


1001
P_7
0001
0101
Bit 4
P_3
P_4
P_8
1000
0000
0100
P_6
P_{10}
Bit 3
P'_5
1010
0010
P_9
0110
Bit 1
Bit 2

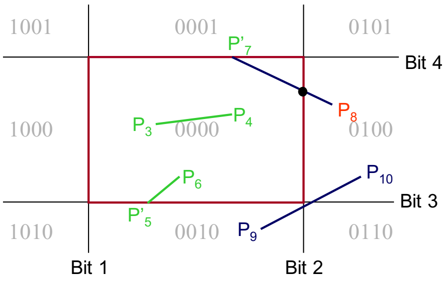
Cohen-Sutherland Line Clipping

- Compute interesections with window boundary for lines that can’t be classified quickly


1001
0001
P'_7
0101
Bit 4
P_3
P_4
P_8
1000
0000
0100
P_6
P_{10}
Bit 3
P'_5
1010
0010
P_9
0110
Bit 1
Bit 2

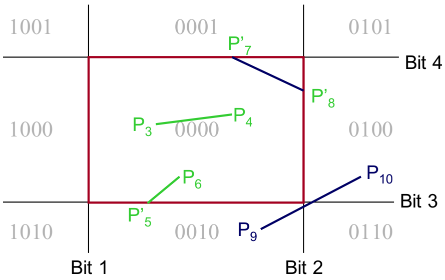
Cohen-Sutherland Line Clipping

- Compute interesections with window boundary for lines that can’t be classified quickly


1001
0001
P'_7
0101
Bit 4
P_3
P_4
P_8
1000
0000
0100
P_6
P_{10}
Bit 3
P'_5
1010
0010
P_9
0110
Bit 1
Bit 2

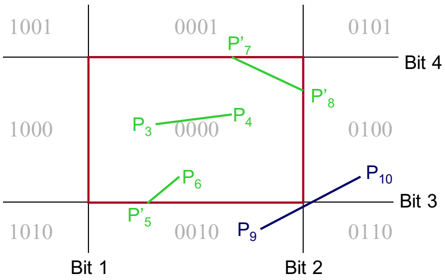
Cohen-Sutherland Line Clipping

- Compute interesections with window boundary for lines that can’t be classified quickly


1001
0001
0101
P'_7
Bit 4
P'_8
1000
P_3
0000
P_4
0100
P_6
P_{10}
Bit 3
P'_5
1010
0010
P_9
0110
Bit 1
Bit 2

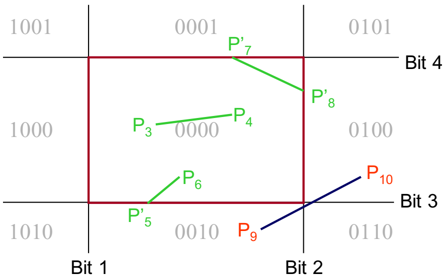
Cohen-Sutherland Line Clipping

- Compute interesections with window boundary for lines that can’t be classified quickly


1001
0001
0101
P'_7
Bit 4
P'_8
1000
P_3
0000
P_4
0100
P_6
P_{10}
Bit 3
P'_5
1010
0010
P_9
0110
Bit 1
Bit 2

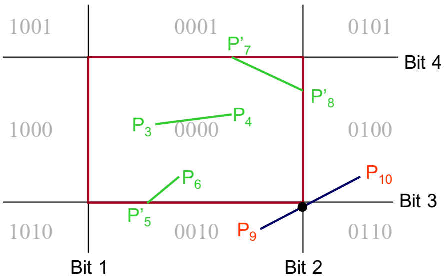
Cohen-Sutherland Line Clipping

- Compute interesections with window boundary for lines that can’t be classified quickly


1001
0001
0101
P'_7
Bit 4
P'_8
1000
P_3
0000
P_4
0100
P_6
P_{10}
Bit 3
P'_5
1010
0010
P_9
0110
Bit 1
Bit 2

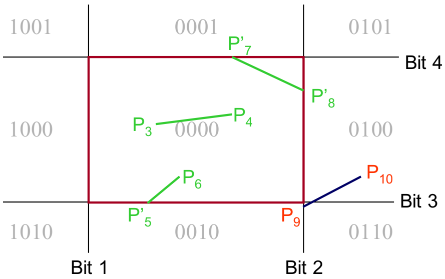
Cohen-Sutherland Line Clipping

- Compute interesections with window boundary for lines that can’t be classified quickly


1001
0001
0101
P'_7
Bit 4
P'_8
1000
P_3
0000
P_4
0100
P_6
P_{10}
Bit 3
P'_5
1010
0010
P_9
0110
Bit 1
Bit 2

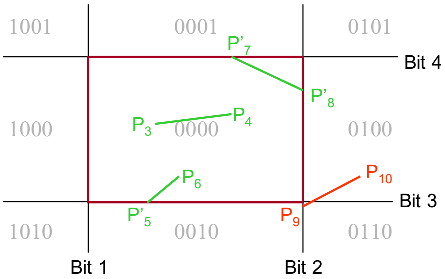
Cohen-Sutherland Line Clipping

- Compute interesections with window boundary for lines that can’t be classified quickly


1001
0001
0101
P'_7
Bit 4
P'_8
1000
P_3
0000
P_4
0100
P_6
P_{10}
Bit 3
P'_5
P_9
1010
0010
0110
Bit 1
Bit 2

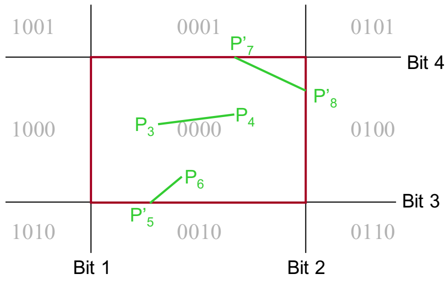
Cohen-Sutherland Line Clipping

- Compute interesections with window boundary for lines that can’t be classified quickly


1001
0001
0101
P'_7
Bit 4
P'_8
1000
P_3
0000
P_4
0100
P_6
P_{10}
Bit 3
P'_5
P_9
1010
0010
0110
Bit 1
Bit 2

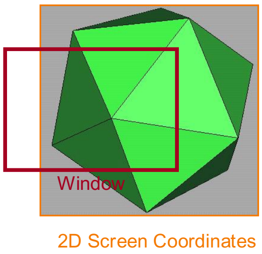
Cohen-Sutherland Line Clipping

- Compute interesections with window boundary for lines that can’t be classified quickly


1001
0001
0101
P’$_7$
Bit 4
P’$_8$
1000
P$_3$
0000
P$_4$
0100
P$_6$
Bit 3
P’$_5$
1010
0010
0110
Bit 1
Bit 2

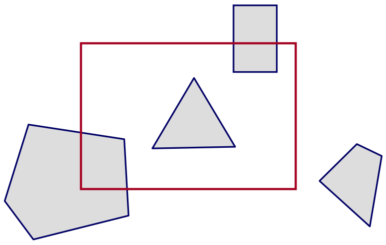
Clipping

- Avoid drawing parts of primitives outside window
  - Points
  - Lines
  - Polygons
  - Circles
  - etc.


Window
2D Screen Coordinates

Polygon Clipping

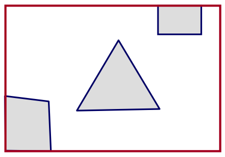
- Find the part of a polygon inside the clip window?



Before Clipping

Polygon Clipping

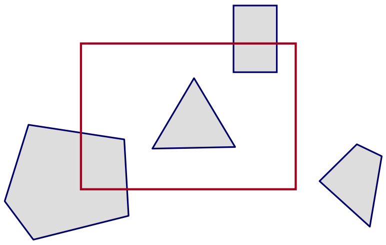
- Find the part of a polygon inside the clip window?



After Clipping

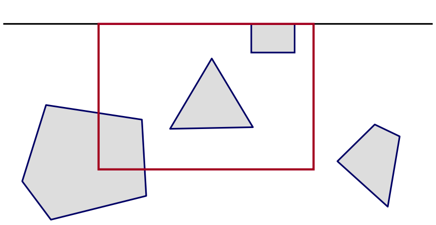
Sutherland Hodgeman Clipping

- Clip to each window boundary one at a time



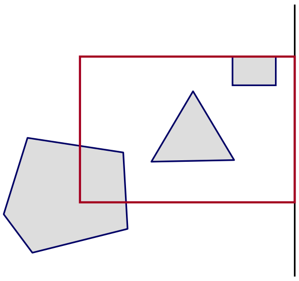
Sutherland Hodgeman Clipping

- Clip to each window boundary one at a time



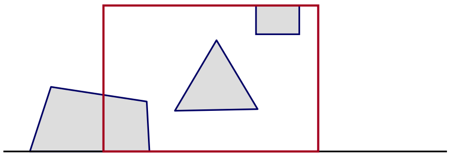
Sutherland Hodgeman Clipping

- Clip to each window boundary one at a time



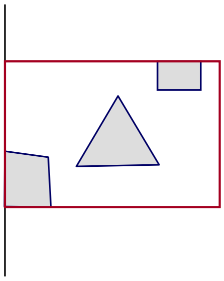
Sutherland Hodgeman Clipping

- Clip to each window boundary one at a time



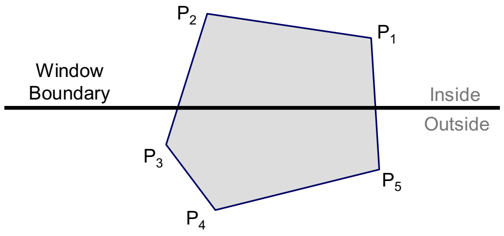
Sutherland Hodgeman Clipping

- Clip to each window boundary one at a time



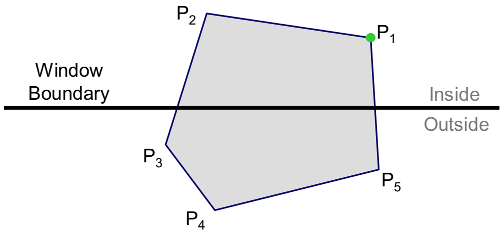
Clipping to a Boundary

- Do inside test for each point in sequence,
  Insert new points when cross window boundary,
  Remove points outside window boundary


P_2
P_1
Window Boundary
Inside
Outside
P_3
P_5
P_4

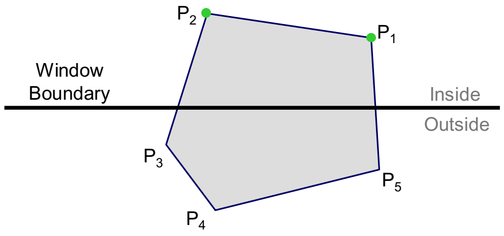
Clipping to a Boundary

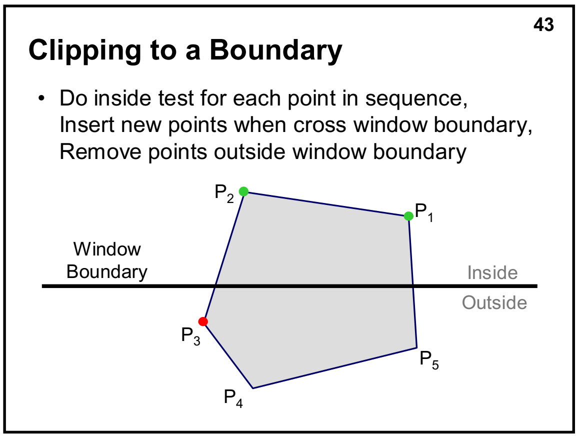
- Do inside test for each point in sequence,
  Insert new points when cross window boundary,
  Remove points outside window boundary


P_2
P_1
Window Boundary
Inside
Outside
P_3
P_5
P_4

Clipping to a Boundary

- Do inside test for each point in sequence,
  Insert new points when cross window boundary,
  Remove points outside window boundary

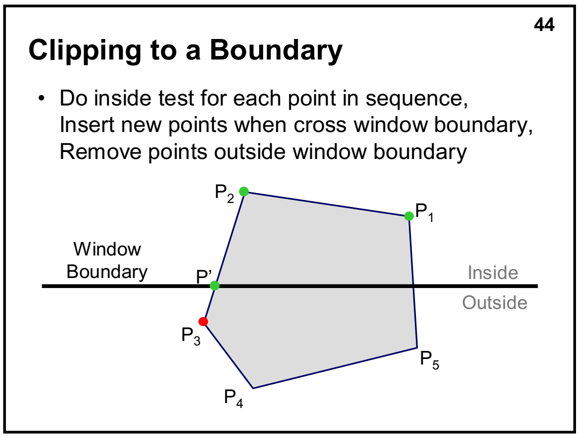

P_2
P_1
Window Boundary
Inside
Outside
P_3
P_5
P_4



Clipping to a Boundary

- Do inside test for each point in sequence, Insert new points when cross window boundary, Remove points outside window boundary

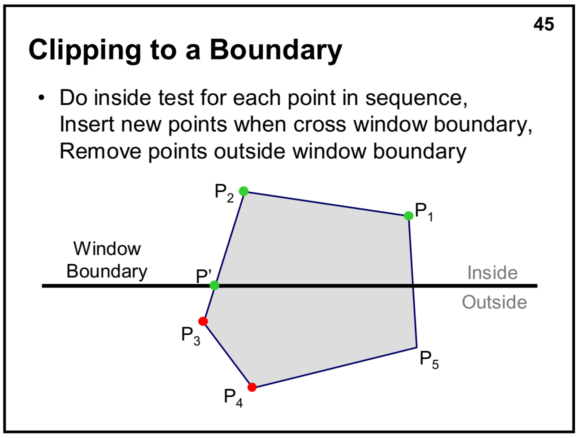
P_2
P_1
Window Boundary
Inside
Outside
P_3
P_5
P_4



Clipping to a Boundary

- Do inside test for each point in sequence, Insert new points when cross window boundary, Remove points outside window boundary

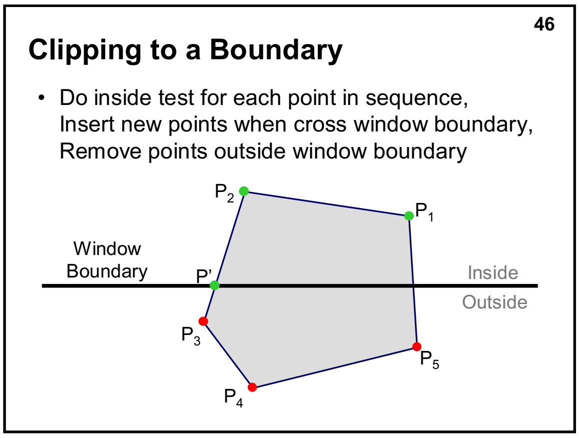
P_2
P_1
Window Boundary
P'
Inside
Outside
P_3
P_5
P_4



Clipping to a Boundary

- Do inside test for each point in sequence, Insert new points when cross window boundary, Remove points outside window boundary

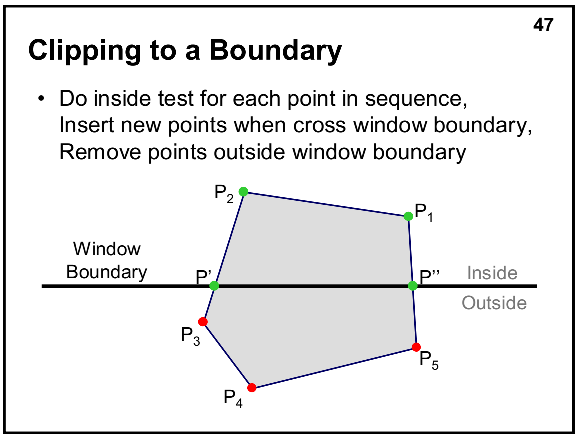
P_2
P_1
Window Boundary
P'
Inside
Outside
P_3
P_5
P_4



Clipping to a Boundary

- Do inside test for each point in sequence, Insert new points when cross window boundary, Remove points outside window boundary

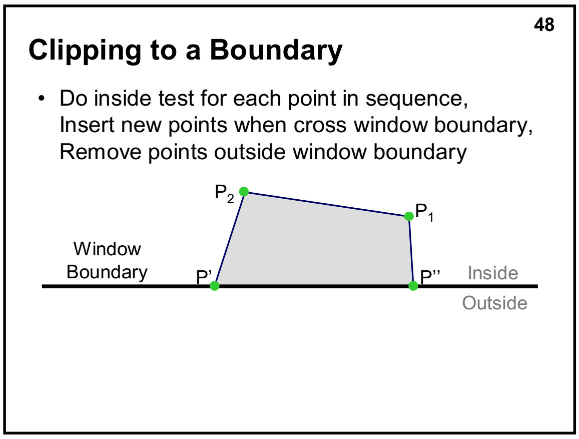
P_2
P_1
Window Boundary
P'
Inside
Outside
P_3
P_5
P_4



Clipping to a Boundary

- Do inside test for each point in sequence, Insert new points when cross window boundary, Remove points outside window boundary

P_2
P_1
Window Boundary
P'
P''
Inside
Outside
P_3
P_5
P_4



Clipping to a Boundary

- Do inside test for each point in sequence, Insert new points when cross window boundary, Remove points outside window boundary

P_2
P_1
Window Boundary
P'
P''
Inside
Outside

2D Rendering Pipeline

3D Primitives

2D Primitives

Clipping — Clip portions of geometric primitives residing outside the window

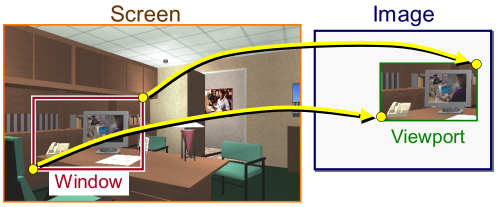
Viewport Transformation — Transform the clipped primitives from screen to image coordinates

Scan Conversion — Fill pixels representing primitives in screen coordinates

Image

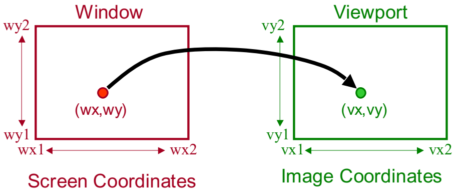
Viewport Transformation

- Transform 2D geometric primitives from screen coordinate system (normalized device coordinates) to image coordinate system (pixels)



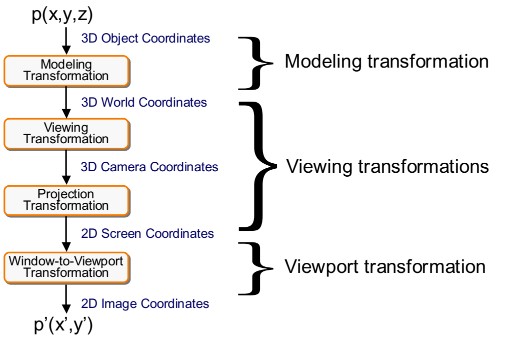
Viewport Transformation

- Window-to-viewport mapping



```
vx = vx1 + (wx - wx1) * (vx2 - vx1) / (wx2 - wx1);
vy = vy1 + (wy - wy1) * (vy2 - vy1) / (wy2 - wy1);
```

Summary of Transformations



p(x,y,z)

3D Object Coordinates

Modeling Transformation — Modeling transformation

3D World Coordinates

Viewing Transformation

3D Camera Coordinates

Projection Transformation — Viewing transformations

2D Screen Coordinates

Window-to-Viewport Transformation — Viewport transformation

2D Image Coordinates

p'(x',y')

Summary

3D Primitives

2D Primitives

Clipping — Clip portions of geometric primitives residing outside the window

Viewport Transformation — Transform the clipped primitives from screen to image coordinates

Scan Conversion — Fill pixels representing primitives in screen coordinates

Image

Summary

3D Primitives

3D Modeling Coordinates

Modeling Transformation

3D World Coordinates

Lighting

3D World Coordinates

Viewing Transformation

3D Camera Coordinates

Projection Transformation

2D Screen Coordinates

Clipping

2D Screen Coordinates

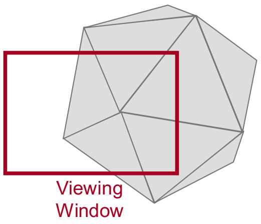
Viewport Transformation

2D Image Coordinates

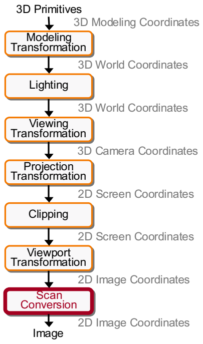
Scan Conversion

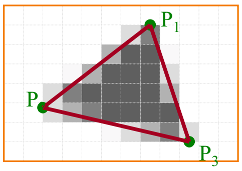
2D Image Coordinates

Image



Next Time


3D Primitives
3D Modeling Coordinates
Modeling Transformation
3D World Coordinates
Lighting
3D World Coordinates
Viewing Transformation
3D Camera Coordinates
Projection Transformation
2D Screen Coordinates
Clipping
2D Screen Coordinates
Viewport Transformation
2D Image Coordinates
Scan Conversion
2D Image Coordinates
Image



Scan Conversion!