Clipping

CS418 Computer Graphics
John C. Hart
Polygon Clipping

- Sutherland-Hodgman
- Polygon $ABC$
Polygon Clipping

- Sutherland-Hodgman
- Polygon $ABC$
- Clip left: $A_1 BCA_2$
Polygon Clipping

• Sutherland-Hodgman
• Polygon $ABC$
• Clip left: $A_1BCA_2$
• Clip right: $A_1B_1B_2CA_2$
Polygon Clipping

- Sutherland-Hodgman
- Polygon $ABC$
- Clip left: $A_1BCA_2$
- Clip right: $A_1B_1B_2CA_2$
- Clip bottom: $A_1'B_1'B_2CA_2$
Polygon Clipping

- Sutherland-Hodgman
- Polygon $ABC$
- Clip left: $A_1B_1CA_2$
- Clip right: $A_1B_1'B_2CA_2$
- Clip bottom: $A_1B_1'B_2'CA_2$
- Clip top: $A_1B_1'B_2'C_1C_2A_2$
Concave Clipping

- Sutherland-Hodgman
- Clip segments even if they are trivially rejectible (rejectionable?)
- Outputs a single polygon that appears as multiple polygons
- Reversed edges don’t get filled
Clipping in 3-D (4-D)

- Need to keep depth (z-coordinate) of geometry for visible surface detection
- Generalize oriented screen edge to oriented clipping plane $C = (A, B, C, D)$
Clipping in 3-D (4-D)

- Need to keep depth (z-coordinate) of geometry for visible surface detection.
- Generalize oriented screen edge to oriented clipping plane $C = (A,B,C,D)$.
- Then any homogeneous point $P = (x,y,z,w)^T$ classified as
  - “on” if $CP = 0$
  - “in” if $CP > 0$
  - “out” if $CP < 0$

\[
Ax + By + Cz + D = 0 \\
\uparrow \\
wAx + wBy + wCz + wD = 0
\]
Vertex Pipeline

- **Model Coords**
- **Model Xform**
- **World Coords**
- **Viewing Xform**
- **Viewing Coords**
- **Projection**

**Homogeneous Divide**

- **Still Clip Coords.**
- **Clipping**
- **Clip Coords.**

**Coordinate Systems**

- **Window Coordinates**
- **Window to Viewport**
- **Viewport Coordinates**

**Process Flow**

1. Model Coords → Model Xform
2. Model Xform → World Coords
3. World Coords → Viewing Xform
4. Viewing Xform → Viewing Coords
5. Viewing Coords → Projection
6. Projection → Homogeneous Divide
8. Still Clip Coords. → Clipping
9. Clipping → Clip Coords.
10. Clip Coords. → Window Coordinates
11. Window Coordinates → Window to Viewport
12. Window to Viewport → Viewport Coordinates

**Shader Stages**

- Vertex Shader
- Scan Converter
- Fragment Shader