



**DCGI**

**DEPARTMENT OF COMPUTER GRAPHICS AND INTERACTION**

# APG Homework Assignment II

**Jakub Hendrich, Daniel Meister**

# Outline

---

- Filling
- Z-buffer
- Perspective projection



# Filling

---

- Scanline filling
- Area mode
  - SGL\_POINT, SGL\_LINE, SGL\_FILL
- Triangle, Polygon
  - glBegin, glEnd
    - SGL\_TRIANGLES, SGL\_POLYGON
- Circle, Ellipse, Arc
  - glBegin, glEnd



# Z-Buffer

---

## ■ Context scope

- Depth value for each pixel
- Similar to the color buffer

## ■ Clear

- `glClear` sets depth of each pixel to the maximum value
  - with `GL_DEPTH_BUFFER_BIT` set

## ■ Enable/Disable

- `glEnable`, `glDisable`
  - `GL_DEPTH_TEST`

## ■ Viewport Transformation

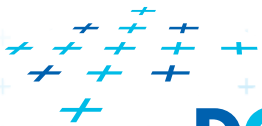
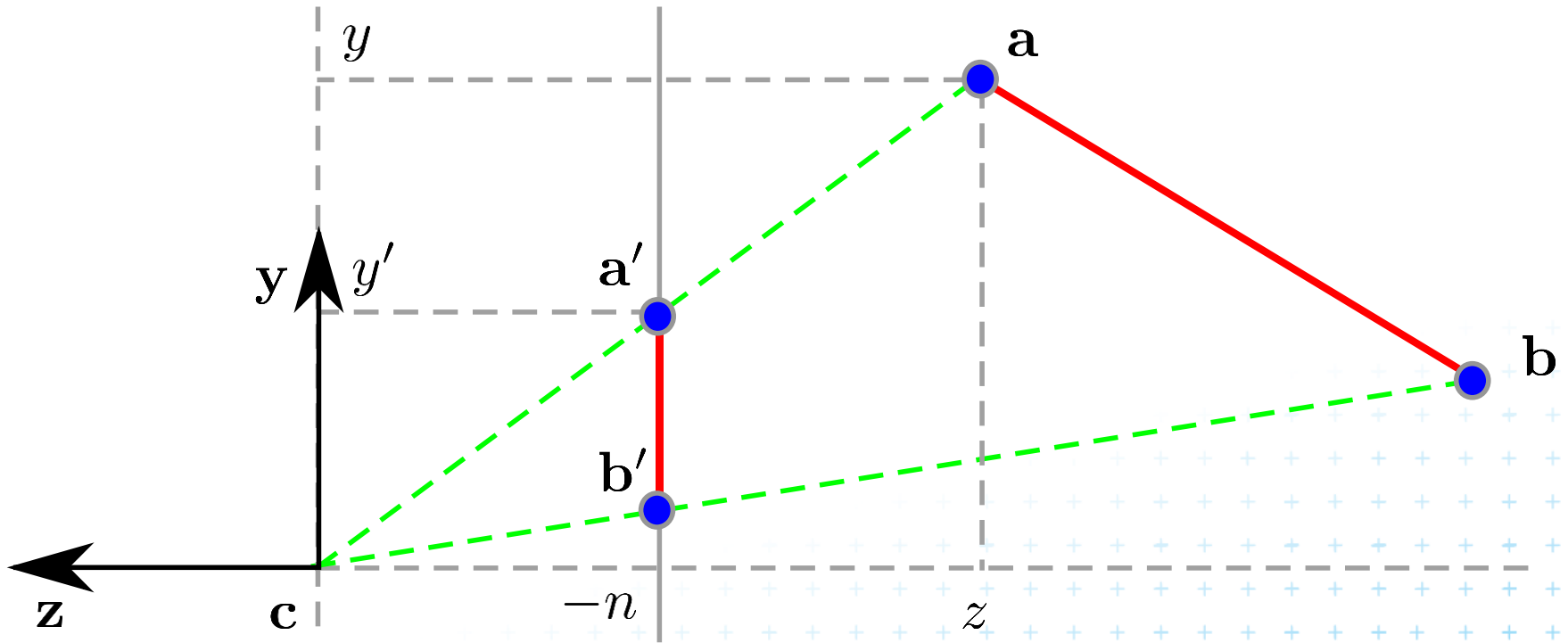
$$\mathbf{V} = \begin{bmatrix} \frac{w}{2} & 0 & 0 & x + \frac{w}{2} \\ 0 & \frac{h}{2} & 0 & y + \frac{h}{2} \\ 0 & 0 & \frac{d}{2} & z + \frac{d}{2} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



# Perspective Projection

$$\frac{x'}{-n} = \frac{x}{z} \Rightarrow x' = -\frac{nx}{z}$$

$$\frac{y'}{-n} = \frac{y}{z} \Rightarrow y' = -\frac{ny}{z}$$



# Perspective Projection

- Depth interpolation

- Bilinear interpolation in the screen space!

- Transformation Stack

- `sglFrustum` multiplies the current matrix by perspective matrix  $\mathbf{P}$

- Color interpolation (1 bonus point)

- Color specified in vertices

$$\mathbf{P} = \mathbf{P}_{\parallel} \mathbf{P}_{\angle} = \begin{bmatrix} \frac{2}{r-l} & 0 & 0 & -\frac{r+l}{r-l} \\ 0 & \frac{2}{t-b} & 0 & -\frac{t+b}{t-b} \\ 0 & 0 & \frac{-2}{f-n} & -\frac{f+n}{f-n} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} n & 0 & 0 & 0 \\ 0 & n & 0 & 0 \\ 0 & 0 & f+n & fn \\ 0 & 0 & -1 & 0 \end{bmatrix}$$



---

# Thank you for your attention!

*Jakub Hendrich*

*21.10.2024*

