Coms 331 **The Projection Matrix** **Fall 2007**

Homework 29

Define the matrices

$$\mathbf{M}_1 = \begin{pmatrix} n & 0 & 0 & 0 \\ 0 & n & 0 & 0 \\ 0 & 0 & -\frac{f+n}{f-n} & -\frac{2fn}{f-n} \\ 0 & 0 & -1 & 0 \end{pmatrix} \text{ and } \mathbf{M}_2 = \begin{pmatrix} \frac{2}{r-l} & 0 & 0 & -\frac{r+l}{r-l} \\ 0 & \frac{2}{t-b} & 0 & -\frac{t+b}{t-b} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

1. Verify that the function $f(x) = a + (b-a)x$ maps the interval $[0, 1]$ into the interval $[a, b]$.

2. Find the inverse of the function in the previous problem. It will map the interval $[a, b]$ into the interval $[0, 1]$

3. Find a linear function that maps the interval $[a, b]$ into the the interval $[-1, 1]$.

4. Let $l = -4, r = 4, b = -3, t = 3, n = 1, f = 10$. Write the matrix $\mathbf{M}_1$ using these values.

5. Use the matrix $\mathbf{M}_1$ found in the previous exercise to find the image of the lower right front corner (r, b, n) of the view frustum.

6. Again using the values $l = -4, r = 4, b = -3, t = 3, n = 1, f = 10$, write the matrix $\mathbf{M}_2$.

7. Use the matrix $\mathbf{M}_2$ found in the previous exercise to map the earlier image of $(r, b, -n)$ to its image under $\mathbf{M}_2$. Is this the lower right front corner $(1, -1, -1)$ of the cube in normalized device coordinates?

8. Multiply the matrices $\mathbf{M}_1$ and $\mathbf{M}_2$ that were found in the earlier exercises, in the order $\mathbf{M}_2\mathbf{M}_1$. This is the projection matrix that OpenGL uses.

9. Starting with the lower right back corner $(r\left(\frac{f}{n}\right), b\left(\frac{f}{n}\right), f)$ and using the same values for r, b, n, and f as before, calculate the image of this point under $\mathbf{M}_2\mathbf{M}_1$. Is it the point $(1, -1, 1)$?

10. The point $(-r, -b, n)$ is *behind* the viewer and to the upper left. (It should not show up in the rendered scene.) Apply the projection transformation $\mathbf{M}_2\mathbf{M}_1$ to this point. What is the result? Is there a problem with that? Would this point be clipped later?

11. The function `glFrustum(l, r, b, t, n, f)` creates the projection matrix $\mathbf{M}_2\mathbf{M}_1$ and multiplies the matrix on the projection stack by it. The function `gluPerspective(fovy, aspect, n, f)` uses its arguments to calculate `l`, `r`, `b`, and `t` and then calls on `glFrustum()` to do the rest of the work. If `fovy` is 90°, `aspect` is 4.0/3.0, `n` is 1.0, and `f` is 10.0, find `l`, `r`, `b`, and `t`, and the calculate the projection matrix.