Computer Graphics
(Comp 4190.410)

Midterm Exam: November 1, 2004

1. (15 points) We want to draw a line $L : ax + by + c = 0$ (of slope $0 < m < 1$) from $x_0$ to $x_n = x_0 + n$, where $a, b, c, x_0$ are integers. (Note that in our textbook we took $a = \Delta y$ and $b = -\Delta x$.) The corresponding y values at the two end points may not have integer values. Let $y_0 = \text{round}(-\frac{ax_0 + c}{b})$ and take $(x_0, y_0)$ as the starting point for a modified Bresenham algorithm. Then we test the sign of

$$F(x_0 + 1, y_0 + \frac{1}{2}) = 2a(x_0 + 1) + 2b(y_0 + \frac{1}{2}) + 2c = 2(ax_0 + by_0 + c) + 2a + b,$$

and may proceed as in the conventional Bresenham algorithm. However, the point $(x_0, y_0)$ is not located on the line $L$; hence, $ax_0 + by_0 + c \neq 0$. Does the algorithm still work? Otherwise, explain why the modified algorithm doesn’t work by giving a counter example or using counter arguments.

2. (20 points)

(a) (10 points) Consider a rotation $R_1$ about an axis $(1, 0, 0)$ by angle $90^\circ$ and another rotation $R_2$ about an axis $(0, 1, 0)$ by angle $120^\circ$. What is the axis and angle of the composite rotation $R_2R_1$?

(b) (10 points) Given a sequence of unit quaternions $q_1, \cdots, q_n$, the product $q_1 \cdot q_2 \cdots q_n$ represents a composite 3D rotation in the local coordinate and the product $q_n \cdot q_{n-1} \cdots q_2 \cdot q_1$ represents a composite 3D rotation in the global coordinate. Explain why this is the case.

3. (15 points) Construct a 2D perspective transformation

$$M = \begin{bmatrix}
    a_{11} & a_{12} & a_{13} \\
    a_{21} & a_{22} & a_{23} \\
    a_{31} & a_{32} & a_{33}
\end{bmatrix}$$

so that it sends 2D points $(0, 0, 1), (1, 0, 1), (0, 1, 1), (1, 1, 1)$ (in homogeneous coordinates) to 2D points $(0, 0, 1), (1, 0, 1), (0, 1, 1), (2, 3, 1)$, respectively.

4. (15 points) In our textbook, the authors claim that the Nicholl-Lee-Nicholl algorithm can be applied only to two-dimensional clipping. What is the main difficulty in extending the NLN algorithm to three-dimensional clipping? How can you reduce the three-dimensional clipping problem to two subproblems of two-dimensional clipping? How can you compare the overall performance of this approach with other algorithms?
5. (40 points)

(a) (10 points) Given two parallel planes \( L_1 \) and \( L_2 \) (with \( d_1 > d_2 \))

\[
L_1 : ax + by + cz + d_1 = 0, \\
L_2 : ax + by + cz + d_2 = 0.
\]

set up two linear inequality conditions for clipping a 3D point \( P(x, y, z) \) with respect to the infinite volume bounded by the two planes.

(b) (15 points) Construct an affine transformation that sends the plane \( L_1 \) to \(-1\) and the other plane \( L_2 \) to \(1\).

(c) (15 points) Using the above result and the 3D viewing transformation based on the line geometry, discuss how to transform an arbitrary 3D view volume to a normalized view volume.