1. [20 points] Consider the following OpenGL commands:

```c
glMatrixMode(GL_MODELVIEW);
glLoadIdentity();
glRotatef( 90.0, 0.0, 1.0, 0.0 );
glTranslatef( 2.0, 0.0, 0.0 );
glScalef( 2.0, 1.0, 1.0 );
```

What will the $4 \times 4$ modelview matrix be equal to after these commands have executed?
3. [10 points] Give the definition of linear transformation.

4. [20 points] Let $f : \mathbb{R}^2 \to \mathbb{R}^2$ be defined to be the transformation that reflects points across the line $y = x - 1$. In particular, it maps a “F” as shown in the picture below.

Give the matrix that represents $f$ over homogeneous coordinates.
1. [20 points] Let \( f : \mathbb{R}^2 \rightarrow \mathbb{R}^2 \) be defined to be the affine transformation that maps an “F” as shown in the picture below.

\[
\begin{array}{c}
\begin{array}{c}
\text{x-axis} \\
\downarrow \\
\langle 0, -1 \rangle \\
\end{array} \\
\begin{array}{c}
\text{y-axis} \\
\uparrow \\
\langle 1, 1 \rangle \\
\end{array}
\end{array}
\]

\( f \) maps to

\[
\begin{array}{c}
\begin{array}{c}
\text{x-axis} \\
\downarrow \\
\langle 2, 0 \rangle \\
\end{array} \\
\begin{array}{c}
\text{y-axis} \\
\uparrow \\
\langle 3, 1 \rangle \\
\end{array}
\end{array}
\]

a. Is \( f \) a rigid transformation? Explain why or why not.

b. Express \( f \) in the form \( f(x) = Mx + b \) with \( M \) a 2 \( \times \) 2 matrix.

c. Give a sequence of “pseudo” OpenGL commands that will draw the “F” in the position shown on the right. Use commands such as `drawF()` (draws “F” in the position shown on the left), `pglRotatef(· · ·)`, `glTranslatef(· · ·)`, `glLoadIdentity()`, and `pglScale2f(· · ·)`.
2. [15 points] Consider the following $3 \times 3$ matrix $M$ that operates on the homogeneous coordinates of points in $\mathbb{R}^2$.

$$
\begin{pmatrix}
-2 & -2 & 2 \\
4 & 0 & 0 \\
0 & 0 & 2
\end{pmatrix}.
$$

In the empty graph on the right, draw the image of the “F” under the affine map on $\mathbb{R}^2$ that is defined by the matrix $M$. Draw to scale, and label points as needed.
1. [20 points] This problem concerns transformations in $\mathbb{R}^3$. Suppose you are given a function \texttt{DrawCone()} that draws a cone of height 1, and base radius 1. This cone drawn by \texttt{DrawCone()} is situated centered around the $y$-axis with its base on the $xz$ plane and the tip of the cone at $(0,1,0)$.

a. Give a code fragment that will draw the cone as shown in the figure: the cone is to be drawn upside down, and with height 2 and base radius 2. Its tip is now at the origin; it is still centered around the $y$-axis.

Your code fragment that draws the cone may use any of the following pseudo-OpenGL commands: \texttt{glMatrixMode()}, \texttt{glLoadIdentity()}, \texttt{glRotatef()}, \texttt{glTranslatef()}, \texttt{glLoadMatrix()}, \texttt{glMultMatrix()}, \texttt{glScalef()}, and \texttt{DrawCone()}.

b. Give a $4 \times 4$ homogeneous matrix that gives the same transformation as is used in your answer for part a.
1. [20 points] This problem concerns transformations in $\mathbb{R}^2$. Suppose you are given a function `DrawCircle()` that draws a unit circle centered at the origin (radius equals one). Give a code fragment that will draw an ellipse as shown in the figure. The length of the ellipse is $\ell$ and the width is $w$. One endpoint of the ellipse is at $(x_0, y_0)$ in $\mathbb{R}^2$, namely, one of the endpoints of the axis along which the length $\ell$ is measured. The ellipsoid is tilted at an angle $\theta$ (measured in degrees).

Your code fragment that draws the ellipse may use any of the following pseudo-OpenGL commands: `glMatrixMode()`, `glLoadIdentity()`, `glTranslatef()`, `glRotatef()`, `glLoadMatrix()`, `glMultMatrix()`, `glScalef()`, and `DrawCircle()`.

3. [20 points] A light source is placed at the origin in $\mathbb{R}^3$, and it casts shadows onto the plane defined by $z = -10$. Thus, the plane is like an infinite wall parallel to the $xy$-plane, placed at $z = -10$.

For $x = (x_1, y_1, z_1)$ a point in $\mathbb{R}^3$ where $z_1 < 0$, let $A(x) = (x_2, y_2, z_2)$ be the point on the wall where the shadow of $x$ is. This means that $z_2 = -10$. Give a $4 \times 4$ matrix that represents the transformation $A$ over homogenous coordinates, or, prove that there is no such matrix.
1. [36 points] Let $f : \mathbb{R}^2 \to \mathbb{R}^2$ be the affine transformation that maps an “F” as shown in the picture below.

```
\begin{tabular}{ll}
\langle -1, 1 \rangle & \langle 1, 1 \rangle \\
\langle 1, 0 \rangle & \langle 0, 0 \rangle \\
\langle 0, -1 \rangle & \langle 0, 1 \rangle
\end{tabular}
```

\[ f \Rightarrow \begin{tabular}{ll}
\langle -1, 1 \rangle & \langle 1, 1 \rangle \\
\langle 1, 0 \rangle & \langle 0, 0 \rangle \\
\langle 0, 0 \rangle & \langle 1, 0 \rangle
\end{tabular} \]

a. Is $f$ a linear transformation?

b. Is $f$ a rigid transformation?

c. Is $f$ orientation preserving?

d. Express $f$ in the form $f(x) = Mx + b$ with $M$ a $2 \times 2$ matrix.

e. Now consider the inverse $f^{-1}$ of the transformation $f$. Give a $3 \times 3$ matrix $N$ that represents $f^{-1}$ in homogeneous coordinates.

f. Express $f$ as a generalized rotation $f = R_u^\theta$ in $\mathbb{R}^2$ by giving the rotation angle $\theta$ and the center point $u$ of the generalized rotation, or explain why this is not possible.
3. [12 points] Suppose the function `drawTwoPoints()` draws a point at \(0, 0, 0\) and another point at \(1, 1, 0\).

a. Consider the sequence of OpenGL commands:

```c
glLoadIdentity();
glTranslatef(-1, 0, 0);
glScalef(2, 1, 2);
glRotatef(90, 0, 1, 0);
drawTwoPoints();
```

When the `drawTwoPoints()` is called, where does the point it draws at \(0, 0, 0\) actually get placed (as transformed by the ModelView matrix)? And, where does the point it draws at \(1, 1, 0\) get placed?

b. Now consider the slightly different sequence of OpenGL commands:

```c
glLoadIdentity();
glRotatef(90, 0, 1, 0);
glScalef(2, 1, 2);
glTranslatef(-1, 0, 0);
drawTwoPoints();
```

When the `drawTwoPoints()` is called, where does the point it draws at \(0, 0, 0\) actually now get placed? (You only need to answer about this one point.)
4. [12 points] Give short answers to the following questions about the Painter’s algorithm:

a. What is the purpose the Painter’s algorithm? What problem does it help solve?

b. Give a short description of how the Painter’s algorithm works.

c. Give an example of how the Painter’s algorithm can fail to accomplish its purpose.