Use the functions \( \text{RotateX}(\text{degrees}) \), \( \text{RotateY}(\text{degrees}) \), \( \text{RotateZ}(\text{degrees}) \), \( \text{Translate}(x, y, z) \), and \( \text{Scale}(sx, sy, sz) \) as found in mat.h.

1. Using the project “cube” as a starting point, create a “3D version” of Homework 2.

   x: apply rotation 30 degrees ccw about x-axis
   y: apply rotation 30 degrees ccw about y-axis
   z: apply rotation 30 degrees ccw about z-axis

   X: apply rotation 30 degrees ccw about an axis through cube’s center, parallel to x-axis
   Y: apply rotation 30 degrees ccw about an axis through cube’s center, parallel to y-axis
   Z: apply rotation 30 degrees ccw about an axis through cube’s center, parallel to z-axis

   s: apply scale factor (0.8, 0.8, 0.8)
   S: apply scale factor (1.2, 1.2, 1.2)

   v: “zoom out” by increasing the size of the clipping volume by 0.2 units each direction
   V: “zoom in” by decreasing the size of the clipping volume by 0.2 units each direction

   t: translate by (0.2, 0, 0)
   T: translate by (-0.2, 0, 0)

   h: shear in the x direction by 0.2y
   H: shear in the x direction by -0.2y

2. a) Write a function \( \text{Rotate}(\text{float degrees}, \text{vec3 direction}) \) that returns a matrix to perform a rotation about an axis through the origin described by the given vector. This is really the same as #5 from homework 3, except you have to convert to spherical coordinates first to get the angles.

b) Write a similar function \( \text{Rotate}(\text{float degrees}, \text{vec3 direction}, \text{vec3 point}) \) in which the rotation is about an axis through the given point in the direction of the given vector.

3. The “simple_spinning_cube” project is an animation of a colored cube rotating about the vertical axis. Using this project as a starting point, do two things:

   (i) Add controls to move the axis of rotation by adjusting the x, y, and z components of the direction vector (which is initially \([0, 1, 0]^{T}\) ):
x: add 0.1 to x
X: add -0.1 to x
y: add 0.1 to y
Y: add -0.1 to y
z: add 0.1 to z
Z: add -0.1 to z

(ii) Draw the axis of rotation.

3. a) Modify the code from (3) so that the cube rotates in a circular orbit in the x-z plane, of radius 3, about the origin. The rotation speed for the orbit should be a tenth that of the cube’s own rotation. (Include a ‘v’ control as in problem #1 for enlarging the view volume.)

b) Add a second cube, same as the first but scaled to one-fifth the size, that orbits around the first one.

(If you are successful in completing both #2 and #3, you can just turn in one project, otherwise you might want to keep them separate.)

**Reminders about spherical coordinates**

![Spherical and rectangular coordinates diagram]

**Spherical to rectangular:**

\[
\begin{align*}
y &= r \cos \theta \\
r' &= r \sin \theta \\
x &= r' \sin \phi \\
z &= r' \cos \phi
\end{align*}
\]

**Rectangular to spherical:**

If x, y, and z are all zero, then phi and theta are undetermined.
If $x$ and $z$ are both zero, then $\phi$ is undetermined and $\theta$ is zero.

If $x$ and $y$ are not both zero, then $\phi$ is $\arctan(x/z)$, and $\theta$ is $\arccos(y/r)$.

To compute the arctangent, it is easiest to use the C function $\text{atan2}(x, z)$, which will give the correct answer even when $z$ is zero. The arctangent returns a value in the range $[-\pi, \pi]$ so adjust accordingly and convert to degrees as needed.

You can get $r$ using the $\text{length()}$ function from vec.h (which is just the distance formula).