**Virtual Trackball Rotation**

To keep track of the trackball and use it to model the rotation of the object I largely followed the slides. This was definitely the hardest part of the project because you have to use dot and cross products. The mouseButton method checks if the left button is being held, starts tracking the movement of the mouse, and calculates and normalizes the position of the mouse when it is clicked. The mouseMotion method checks if the mouse is being tracked, and if it is calculates the x,y, and z components of the mouse’s current position. These values are used along with the last position values to calculate the change in position. This change in position is used to calculate the rotation angle and the cross product is used to calculate the rotation axes. The last part of the mouseMotion method is to update the position of the mouse.

One line of code I still am struggling to understand is the

**d = (d < 1.0f) ? d : 1.0f;**

It is from the PowerPoint that we looked at in class and I know that it has something to do with making sure that the z value is on the actual trackball surface. I just don’t understand what the question mark and colon do.

**Translation**

To implement the translation, I used the Specialkey method so that the arrow keys can be used. Each press of the arrow key will move the object over by 0.5 in that direction. I had some issues with the translation because I was not resetting the translation after every pass over the matrix stack. After doing some research, I found that I could fix this by creating two global variables, x\_TotalTrans and y\_TotalTrans to keep track of how much the object has been translated overall. This seems to have fixed the issues I was having.

**Zoom In/Out**

I implemented the zoom in and zoom out feature to zoom in and out using the ‘i’ and ‘o’ keys. When I first implemented the zoom, I made it so that each press zooms in at 0.1 but it looked very choppy so I changed it to 0.5.

BONUS SCREENSHOT ON NEXT PAGE

**BONUS**

A screenshot of a computer screen

Description automatically generated