CMSC427 Notes on Project 2
Virtual Trackball

All the ideas for project 2 are in the project description (PD), and in the trackball tutorial (TT), on Elms. Here's how to read those documents to figure out the project.

Step 1: Implement a mouse listener to get the mouse position as it drags. You'll need to save the last position (m0) to compare to the current (m1). See Section 1 of the PD for info. You can find other tutorials online on Java mouse listeners. From this you get two 2D points m0 and m1.

Step 2: Map the 2D (x,y) mouse positions to 3D positions on the virtual trackball. Switch to the TT, page 3, bottom of the page for formulas and code for this. While in the TT you compute a vector named z, you use this method to compute the vectors u and v for the mouse positions m0 and m1.

Step 3: Now you need to compute the angle-axis representation of a rotation from the mouse position m0 (and u) to the mouse position m1 (and v). There's a Matrix4f method that gives you the rotation matrix directly from the angle and axis.

The TT uses quaternions to do the rotation. We will cover quaternions, but we can manage this with angle axis. Switch back to the PD for the third paragraph in section 1, and there's two sentences that start "Use the cross product …" that gives exactly how to compute the rotation matrix you need. The code here is short – about three lines.

Step 4: Take care of special cases, as noted in the PD. That's in the TT in section 3, other details.

We have done almost all the math required for this project. We have not intersecting a ray with a sphere, but the code for that is given in the TT along with the equations.

Your final code should not be long – if it's more than a page, you may be off track (pun intended …)

This project is offered at other schools. This presentation gives some good diagrams to under the projection of x,y to the x,y,z on the sphere.

http://web.cse.ohio-state.edu/~shen.94/781/Site/Slides_files/trackball.pdf

There is code online for virtual trackballs, and my expectation is that you try to solve the problem yourself, if you get stuck look at the code online, but don't copy it.