SUMMARY

The assignment covered basics of texture mapping.The 3D rotations of the camera where achieved using unit quaternion multiplications. Two ply models were rendered in the scene. The normal per vertex was computed as the average of normals of the faces that share the vertex. A third party ply file reader was used. Three light sources were used in the scene.The code implementation greatly follows the MVC paradigm and extensive use of OOP concepts. Intermediate mapping techniques were used. Spherical intermediate mapping and cylindrical intermediate mapping were used on the models. The texture was read from a bitmap image and s,t parameters where computed using coordinate system transformations and trignometric concepts. Dynamic texturing was achieved using stencil buffering.

Lessons learnt:

1.Use of quaternions over normal Rotatef function.
2.OOP concept and MVC architecture.
3.In depth knowledge about C++ concepts.
4.Trackball implementation.
5.ply files and usages.
6.Memory leaks in C++.
7.Intermediate texture mapping techniques.
8.BMP image pros and cons.
9.OpenGL lighting.
10. Stencil buffer for object picking.

References:

1.http://www.stackoverflow.com/ -for debugging
2.http://www.wikipedia.org/ – for quaternions and arcball implementation
3.opengl_programming_guide_8th_edition- The redbook -for OpenGL library functions
4.Lecture videos by Ken Joy in youtube channel UC Davis Academics
5.http://people.sc.fsu.edu/~jburkardt/data/ply/ - ply files.
6.http://www.turbosquid.com/ - texture files.