

OpenGL and glut Notes

CS 402 – Fall 2005

Dr. Joan Krone

1. For 2D problems, the directions of positive increments in x and y in OpenGL are to the right and up. For input functions used in GLUT and windowing systems, positive increments usually are down and to the right.
2. The form of GLUT callback functions is fixed. So global variables may be necessary to pass values between functions.
3. OpenGL is a state machine: It takes inputs and produces outputs. Between the inputs and the production of outputs, descriptions of objects are converted to images.
4. OpenGL has two types of functions: those that specify inputs to the machine (objects) and those that change the state of the machine. State changing functions include those that specify colors, viewing conditions, material properties, and other variables. The state determines how the inputs are processed.
5. It is necessary to enable features desired for a particular application. Example: `glEnableClientState`. There is a corresponding `glDisableClientState`.
6. Pushes and Pops occur to the stack corresponding to the present matrix mode. Pushes and pops must be paired in a program to be able to return to the desired state.
7. Example: `glMatrixMode(GL_PROJECTION); /*set projection matrix and draw scene */ glPushMatrix(); /*change the projection matrix and draw scene */ glPopMatrix();`
8. GLUT callbacks that return a mouse position give the position in screen coordinates with the origin in the top left corner.
9. `glutKeyboardFunc()` only responds to the pressing of a key and not to its release.
10. The window size should be global so that the mouse position can be converted correctly to world coordinates.
11. Polygons defined by more than three vertices might not be planar. If you intend them to be, you need to put in a check.
12. The clipping volume set in `glOrtho()` is measured from the origin in eye space. So, the near distance should be less than the far distance.
13. The depth buffer must be enabled and cleared for hidden surface removal.
14. GLU quadrics and GLUT objects are constructed from polygons and may not look smooth if they are not defined with a sufficient number of polygons.
15. The parameters for viewing are in camera or viewing coordinates. The near and far parameters are measured from the camera to the clipping planes. For perspective views, $far > near > 0$.
16. Placing the front clipping plane too close to the camera can lead to numerical errors in depth calculations for perspective views.
17. Parameters in `glTranslate()` represent the distances from a fixed camera at the origin, i.e., a positive dx is a translation to the right of the camera, a positive dy is a translation above the camera, and dz is a distance behind the camera.

18. A positive direction of rotation about a vector is counterclockwise when looking from the head of the vector to its tail.
19. The last transformation specified before drawing any geometry is the first applied. This means that the order in which transformations are applied is the opposite of the order in which they appear in the program.
20. Transformations affect the OpenGL state. Since the state is global, it is important to be careful about changes made to matrices in various parts of the program.
21. Once lighting is enabled, colors assigned by `glColor()` are no longer used.
22. Lighting calculations must be enabled and each light source must be enabled individually. For a single source, we use:
`glEnable(GL_LIGHTING);`
`glEnable(GL_LIGHT0);` These two commands ask OpenGL to do the shading calculations. All colors will be assigned based on light sources and material properties rather than by `glColor()`. Individual lights must be turned on and off separately.
23. Shading for back faces will not be correct unless the light model is set to two-sided: `glLightModel(GL_LIGHT_MODEL_TWO_SIDE, GL_TRUE);`
24. Shading may be unsatisfactory if you use large polygons due to the interpolation of vertex shades across a large area.
25. Lighting calculations require that normals have unit length.
26. Scaling changes the lengths of normals.