GLUT tips + pitfalls

alexandri zavodny

GLUT barebones

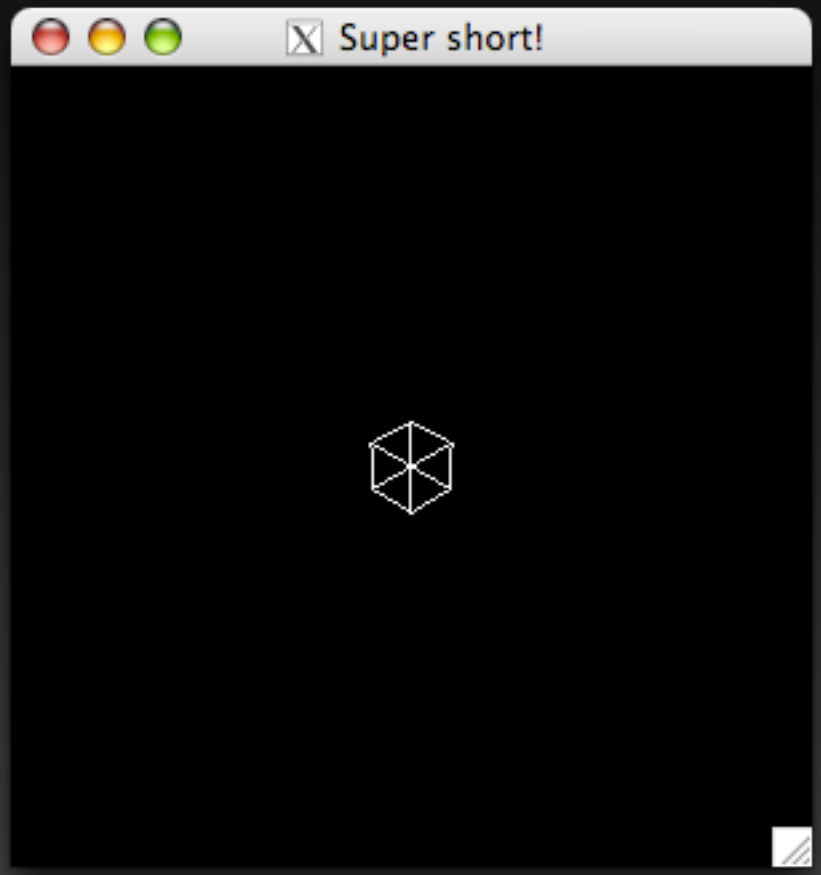
What is the simplest GLUT app you can write?

GLUT barebones

```
 1 #include <GL/glut.h>
 2
 3 void renderScene()
 4 {
 5     glClearColor(0,0,0,1);
 6     glClear(GL_COLOR_BUFFER_BIT);
 7
 8     glMatrixMode(GL_PROJECTION);
 9     glLoadIdentity();
10     gluPerspective(45, 1.0, 0.1, 100);
11
12     glMatrixMode(GL_MODELVIEW);
13     glLoadIdentity();
14     gluLookAt(10,10,10,0,0,0,0,1,0);
15
16     glutWireCube(1.0f);
17
18     glFlush();
19 }
```

```
21 int main(int argc, char **argv)
22 {
23     glutInit(&argc, argv);
24     glutCreateWindow("Super short!");
25
26     glutDisplayFunc(renderScene);
27
28     glutMainLoop();
29
30     return 0;
31 }
```

GLUT barebones



GLUT barebones

What is the simplest GLUT app you'd WANT to write?

GLUT barebones

- What is the simplest GLUT app you'd WANT to write?
- Leave nothing to fate:
  - Set the window size + default position
  - Set the projection and modelview matrices
  - Set the object's color
  - Request double buffering, depth buffer, RGBA color buffer
  - Make sure that depth testing is on (GL_DEPTH_TEST)
  - Clear the color buffer / depth buffer!
  - Swap the buffers / flush to buffer

GLUT barebones

- Not much longer:

```
 1 #include <GL/glut.h>
 2
 3 void renderScene()
 4 {
 5     glEnable(GL_DEPTH_TEST);
 6     glClearColor(0,0,0,1);
 7     glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);
 8
 9     glMatrixMode(GL_PROJECTION);
10     glLoadIdentity();
11     gluPerspective(45, 1.0, 0.1, 100);
12
13     glMatrixMode(GL_MODELVIEW);
14     glLoadIdentity();
15     gluLookAt(10,10,10,0,0,0,0,1,0);
16
17     glColor3f(1,1,1);
18     glutWireCube(1.0f);
19
20     glutSwapBuffers();
21 }
```

```
23 int main(int argc, char **argv)
24 {
25     glutInit(&argc, argv);
26     glutInitWindowSize(512,512);
27     glutInitWindowPosition(100,100);
28     glutInitDisplayMode(GLUT_DOUBLE | GLUT_DEPTH | GLUT_RGBA);
29     glutCreateWindow("Super short!");
30
31     glutDisplayFunc(renderScene);
32
33     glutMainLoop();
34
35     return 0;
36 }
37 _
```

Matrix Maintenance

- Keep track of your GL_PROJECTION and GL_MODELVIEW matrices!
  - glLoadIdentity() when appropriate
- Know your mode / pick a convention
  - Single projection: update in resize callback and forget about it!
  - Multi projection: update every frame!

Matrix Maintenance

- glOrtho / gluOrtho2D
- glTranslate / glRotate / glScale
- glFrustum
- gluPerspective
- gluLookAt

Matrix Maintenance

- Know which functions are appropriate
- GL_PROJECTION
  - glOrtho / gluOrtho2D
  - gluPerspective
  - glFrustum
- GL_MODELVIEW
  - gluLookAt
  - glTranslate / glRotate / glScale

Display Lists

- Why?
  - Efficiency.
- Why not?
  - Too static.
- How?
  - GLuint displaylist = glGenLists(1);
  - glNewList(displaylist, GL_COMPILE);
    - // drawing code
  - glEndList();

Lighting Checklist

- To get lighting:
  - 1) Lighting must be enabled
  - 2) Objects must have materials
    - 2a) Set material properties with glMaterial*()
    - 2b) Override material properties and use glColor()
  - 3) Objects must have normals!
    - All GLUT primitives and GLU quadrics have them yay
    - If you use glScale, use glEnable(GL_NORMALIZE)!

Lighting Checklist (cont'd)

- Step 1: Enabling lighting
  - glEnable(GL_LIGHTING);
- But there are no lights, so…
  - glEnable(GL_LIGHT0); //or 1, 2, … up to 7*
- And that light has no properties, so…
  - glLightfv(GL_LIGHT0, GL_POSITION, position);
  - glLightfv(GL_LIGHT0, GL_DIFFUSE, diffuseColor);
  - glLightfv(GL_LIGHT0, GL_AMBIENT, ambientColor);

*Some versions of OpenGL have fewer!

Lighting Checklist (cont'd)

- Step 2: Use materials or setup color tracking
  - If using materials:
    - glColor won't do anything ; use glMaterialfv() instead
    - Example:
      - Glfloat red[] = {1, 0, 0, 1};
      - glMaterialfv(GL_FRONT_AND_BACK, GL_AMBIENT_AND_DIFFUSE, red);
  - If using color tracking:
    - glEnable(GL_COLOR_MATERIAL);
    - glColorMaterial(GL_FRONT, GL_AMBIENT_AND_DIFFUSE);

Lighting Checklist (cont'd)

- Step 3: Normals
  - GLUT primitives & GLU quadrics have them
  - Inside of GL_QUADS, GL_TRIANGLES, etc.:
    - Call glNormal3f(nx, ny, nz) before each glVertex3f.
- Step 4: PROFIT

Misc Lighting Tips

- When using glMaterialfv or glLightfv, make sure your arrays have 4 elements!!
  - For light position, the last component should be 1
- Light position gets transformed by the modelview matrix just like a vertex!

Misc. Tips

- Follow protocol!
- Create a “debug mode” that draws:
  - A grid over the ground plane
  - The axes, showing +/- XYZ directions

Misc. Tips

- All of the gl* functions allow you to *set…* glGet() allows you to get info back!
  - Check if state variables are enabled / disabled
  - Great for debugging
- glGetError() returns OpenGL-specific ‘errno’
- Sketch out your app ahead of time!