Computer Graphics

Z Sweedyk
Lecture 11
10/18/00

Graphics Pipeline –
What does the user do?

1. Build
Primitives
(World Coordinates)

2. Assemble
Scene
(World Coordinates)

3. Clip
(Clipped Coordinates)

4. Normalize
(Normalized Coordinates)

5. Project
(into 2D
(Normalized Coordinates)

6. Scan
Convert

User Defines:
• Objects in the scene and how they are positioned
• Viewing system
• Lighting
• State Variables
• I/O

Defining the Objects:
GL Primitives
• GL_POINTS
• GL_LINES
• GL_TRIANGLES
• GL_POLYGON
• Etc. (see WNDS pp 44-45)

GL Primitives: Example 1

```c
glColor3f(.5,.5,.5);
glBegin(GL_POLYGON);	glVertex2f(0.0,0.0,0.0);	glVertex2f(10.0,0.0,0.0);	glVertex2f(10.0,10.0);	glVertex2f(0.0,10.0);	glEnd();
```

GL Primitives: Example 2

```c
glColor3f(.5,.5,.5);
glRectf(0.0,0.0,10.0,10.0);
```
GL Primitives: Example 3

```c
glBegin(GL_POINTS);
    glColor3f(1.0,0.0,0.0);
    glVertex2f(0.0,0.0);
    glVertex2f(10.0,0.0);
    glColor3f(0.0,0.0,1.0);
    glVertex2f(10.0,10.0);
    glVertex2f(0.0,10.0);
glEnd();
```

GL Primitives: Example 4

```c
    glColor3f(1.0,0.0,0.0);
    glutWireCube(10.0);
```

Positioning the Object: GL Transformations

- `glTranslate`
- `glRotate`
- `glScale`

For details see WNDS pp 110-112

Putting It Together

```c
    glColor3f(.5,.5,.5);
    glRotatef(45.0,0.0,0.0,1.0);
    glRectf(0.0,0.0,10.0,10.0);
```

Exercise 1

Build a 5x10x1 box with a blue top/bottom, red left/right, and green front/back. The box should be centered at the origin. You can use the following:
- `glRect(0.0,0.0,1.0,1.0)`
- `glTranslatef(dx,dy,dz)`
- `glScalef(sx,sy,sz)`
- `glRotatef(angle,vx,vy,vz)`
- `glLoadIdentity()`
- `glColor3f(red,green,blue)`

Matrix Stack

- `glPushMatrix()`
- `glPopMatrix()`
Exercise 2

Now rotate your box:
45 degrees about the x axis
then 45 degrees about the y axis

Defining the Viewing System:
Viewpoint

Default:
• Eye Position: (0,1,0)
• Viewing Direction: (0,0,-1)
• Up: (0,1,0)

Viewpoint

• Is it part of the viewing system?
• Is it part of the model?

```c
gluLookAt(GLdouble eyex, GLdouble eyey, GLdouble eyez,
         GLdouble centerx, GLdouble centery, GLdouble centerz,
         GLdouble upx, GLdouble upy, GLdouble upz);
```

Defining the Viewing System:
Projection Mode/Viewing Volume

1. `glOrtho(GLdouble left, GLdouble right, GLdouble bottom, GLdouble top, GLdouble near, GLdouble far);`
2. `glFrustum(GLdouble left, GLdouble right, GLdouble bottom, GLdouble top, GLdouble near, GLdouble far);`

Example

```c
glMatrixMode(GL_PROJECTION);
glLoadIdentity();
glOrtho(-10.0,10.0,-10.0,10.0,1.0,21.0);
glMatrixMode(GL_TRANSFORMATION);
glLoadIdentity();
glColor3f(1.0,0.0,0.0);
glTranslatef(0.0,0.0,-10.0);
glutSolidCube(5.0);
```

Exercise

Establish a viewing system with perspective projection to view your rotated box.
Defining the Viewing System: Viewports

- Default: projection plane maps to window
- glViewport: see WNDS pp 130

Lighting

- Define lights
- Define material properties
- Enable lighting/lights

See WNDS ch 5 for details