

Transform

Shaoting Zhang

May 11, 2008

- My office Hour: Mon. 1:00-3:00pm, Hill-418
- All of the slides and codes can be found by clicking my name in the course's website

- 1 Questions
- 2 HW1
- 3 Transform
- 4 Q/A

Question 1

What's the result?

```
glColor3f(1.0,0.0,0.0);  
glPointSize(50.0);  
glBegin(GL_POINTS);  
    glColor3f(0.0,1.0,0.0);  
    glPointSize(10.0);  
    glVertex2i(0,0);  
glEnd();
```

Result

- According to OpenGL man pages: `GL_INVALID_OPERATION` is generated if `glPointSize` is executed between the execution of `glBegin` and the corresponding execution of `glEnd`.
- Actually in C and Java you will get A **green** point with size **50.0**

Question 2

Which line do we get?

```
glBegin(GL_LINES);  
    glVertex2i(0,0);  
    glVertex2i(1,1);  
    glVertex2i(0,1);  
glEnd();
```

Result

A line from $(0,0)$ to $(1,1)$.

Question 3

Something wrong?

```
for(double f=0.0, f!=1.0; f+=0.1)
```

Result

With the $!=$ it might not be exact, and might skip over 1.0 (and instead land on 1.00000000001 or something).

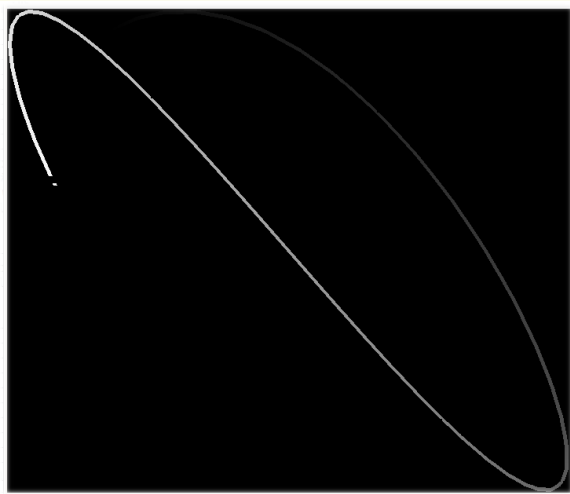
Use $f \leq 1.0$

- 1 Questions
- 2 HW1
- 3 Transform
- 4 Q/A

What do we have?

- 1 draw(GL gl) function, which will be invoked during every interval time
- 2 current time, t , which determine the θ

What's the effect?



Tips

$$x = \sin(a\theta + \delta), y = \sin(b\theta), \theta \in [t, t + \pi]$$

- You need a loop to draw line pieces
- `GL_LINE_STRIP` is a good choice
- For the extra credits, consider the parameters

- 1 Questions
- 2 HW1
- 3 Transform
- 4 Q/A

Draw a rectangle

```
void transform(void)
{
    glClear(GL_COLOR_BUFFER_BIT);
    glColor3f(0.9, 0.9, 0.9);
    glRectf(-50, -50, 50, 50);
}
```

Transform functions

- `glTranslate*(x,y,z)`: produce a translation by vector (x,y,z)
- `glRotate*(angle,x,y,z)`: produce a rotation of *angle* degrees around the vector (x,y,z)
- `glScale*(x,y,z)`: produce a nonuniform scaling along the x, y and z axes.

Translate it

along x axis

```
glColor3f(0.6, 0.6, 0.6);  
glTranslatef(150.0, 0.0, 0.0);  
glRecti(-50, -50, 50, 50);
```

Something wrong?

Note:

We need `glLoadIdentity()` function (or `glPopMatrix()` and `glPushMatrix()`) to restore the transformation matrix to identity matrix. Otherwise the model will be transformed every time when the window is refreshed.

Correct codes

```
void transform(void)
{
    glClear(GL_COLOR_BUFFER_BIT);
    glLoadIdentity();
    glColor3f(0.9, 0.9, 0.9);
    glRectf(-50, -50, 50, 50);
    glColor3f(0.6, 0.6, 0.6);
    glTranslatef(150.0, 0.0, 0.0);
    glRectf(-50, -50, 50, 50);
}
```

Rotate it

Translate it along x axis, then rotate it by z axis

```
glLoadIdentity();  
glColor3f(0.4, 0.4, 0.4);  
glTranslatef(150.0, 0.0, 0.0);  
glRotatef(45.0, 0.0, 0.0, 1.0);  
glRectf(-50, -50, 50, 50);
```

Something wrong?

Note: The current matrix is **postmultiplied** by the matrix specified before, which means, in an OpenGL program a transformation sequence is applied in the opposite order from which it is specified.

Correct codes

Translate it along x axis, then rotate it by z axis

```
glLoadIdentity();  
glColor3f(0.4, 0.4, 0.4);  
glRotatef(45.0, 0.0, 0.0, 1.0);  
glTranslatef(150.0, 0.0, 0.0);  
glRectf(-50, -50, 50, 50);
```

Scale it

Translate, Rotate, Scale

```
glLoadIdentity();  
glColor3f(0.0, 0.0, 0.0);  
glScalef(2.0, 1.0, 1.0);  
glRotatef(90.0, 0.0, 0.0, 1.0);  
glTranslatef(150.0, 0.0, 0.0);  
glRectf(-50, -50, 50, 50);
```

Stack processing

- **glPushMatrix()**: Copy the top matrix in the stack and store copy in the second stack position.
- **glPopMatrix()**: Erase top matrix in stack and move second matrix to top of stack.

Previous example

```
void transform(void)
glPushMatrix();
glRecti(-50, -50, 50, 50);
glTranslatef(150, 0, 0);
glRecti(-50, -50, 50, 50);
glPopMatrix();
glPushMatrix();
glRotatef(45, 0, 0, 1);
glTranslatef(150, 0, 0);
glRecti(-50, -50, 50, 50);
glPopMatrix();
```

General rotate

Rotate along $P(50,0,0) \rightarrow P(50,0,1)$

```
glColor3f(0.9, 0.9, 0.9);  
glRotatef(90.0, 0.0, 0.0, 1.0);  
glRecti(200,-50,300,50);
```

General rotate

Translate, rotate, translate

```
glColor3f(0.4, 0.4, 0.4);  
glTranslatef(150.0, 0.0, 0.0);  
glRotatef(90.0, 0.0, 0.0, 1.0);  
glTranslatef(-150.0, 0.0, 0.0, 1.0);  
glRecti(200,-50,300,50);
```

3D transform - Solar system

Sun

```
glLoadIdentity();  
glColor3f (1.0, 0.0, 0.0);  
glutWireSphere(1.0, 50, 50);
```

3D transform - Solar system

Earth

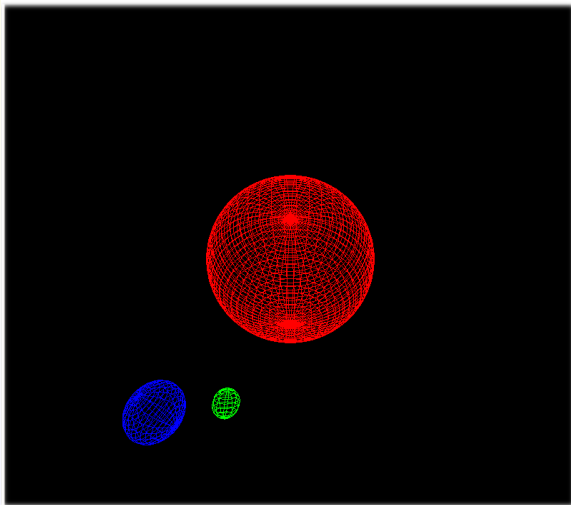
```
glLoadIdentity();  
glRotatef((GLfloat) year, 0.0, 1.0, 0.0);  
glTranslatef (2.0, 0.0, 0.0);  
glColor3f(0.0,0.0,1.0);  
glutWireSphere(0.2, 20, 20);
```

3D transform - Solar system

Moon

```
glLoadIdentity();  
glTranslatef(2.0*cos(year*2*pi/360),0, 2.0*sin(-year*2*pi/360));  
glRotatef((GLfloat) month, 0.0, 1.0, 0.0);  
glTranslatef(-2.0*cos(year*2*pi/360),0, -2.0*sin(-year*2*pi/360));  
glRotatef((GLfloat) year, 0.0, 1.0, 0.0);  
glTranslatef (2.5, 0.0, 0.0);  
glColor3f(0.0,1.0,0.0);  
glutWireSphere(0.1, 10, 10);
```

3D transform - Solar system



- 1 Questions
- 2 HW1
- 3 Transform
- 4 Q/A

Any questions?