

Computer Graphics

Tutorial week 2

Today

- OpenGL basics
- OpenGL assignment
- Raytracer assignment

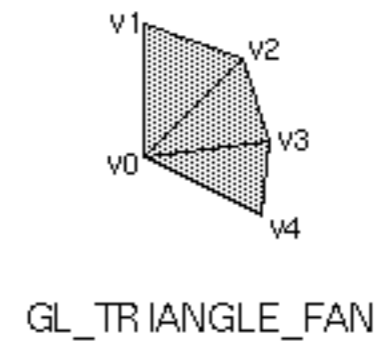
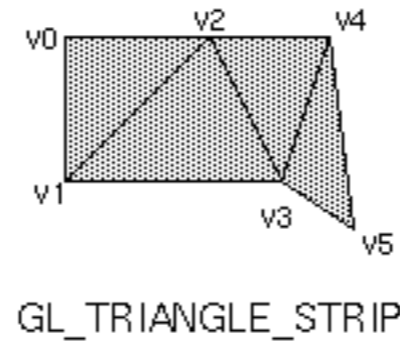
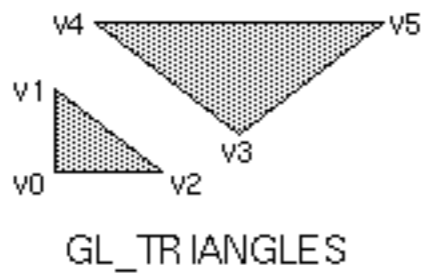
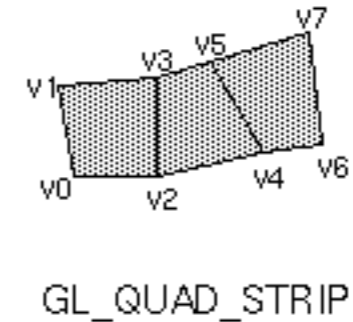
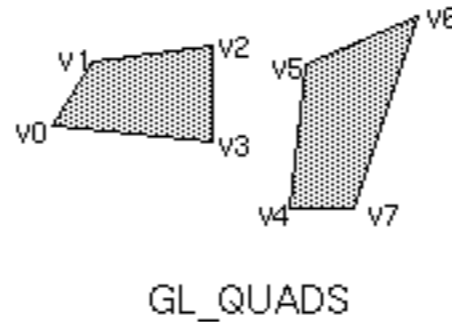
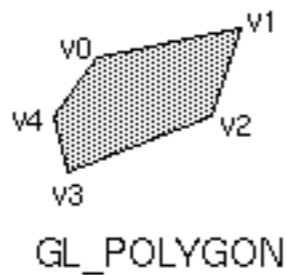
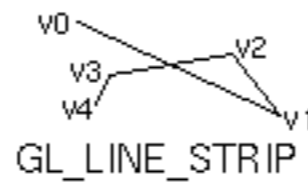
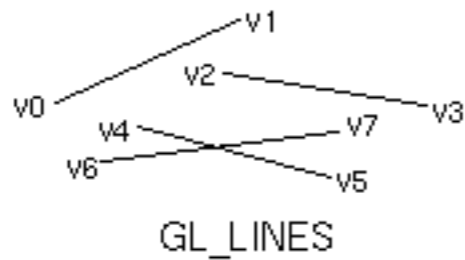
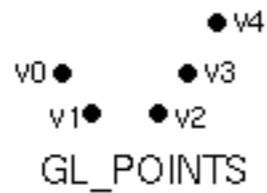
OpenGL basics

Immediate mode

- ‘Old-fashioned’ way of specifying primitives
- glBegin()/glEnd() construction
- Example:

```
glBegin(GL_LINES);  
glVertex3f(1.0, 0.0, 3.0);  
glVertex3f(1.0, 1.0, 8.0);  
glEnd();
```
- Demo (created by Nate Robins)

Drawing primitives



OpenGL transformations

- OpenGL is a right-handed system
- Order of transformations

processed first, T_2

- $P' = T_n \cdot \dots \cdot T_2 \cdot T_1 \cdot P$

only for column x

OpenGL transformations

- **Translation:**

```
glTranslatef(GLfloat x, GLfloat y, GLfloat z);
```

- **Rotation:**

```
glRotatef(GLfloat angle, GLfloat x, GLfloat y, GLfloat z);
```

- **Scaling:**

```
glScalef(GLfloat x, GLfloat y, GLfloat z);
```

- **Demo (created by Nate Robins)**

OpenGL - viewing

- **Perspective projection:**

```
void gluPerspective(GLdouble fovy,  
                  GLdouble aspect,  
                  GLdouble near,  
                  GLdouble far);
```

- **Positioning view point (convenience function):**

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

- **Again: demo (created by Nate Robins)**

Matrix stacks

- Remember: OpenGL is a state-machine
- Useful for constructing hierarchical models
- Push: `void glPushMatrix(void);`
- Pop: `void glPopMatrix();`
- Start from scratch again: `glLoadIdentity();`
- Example: drawing wheels of a car

Display lists

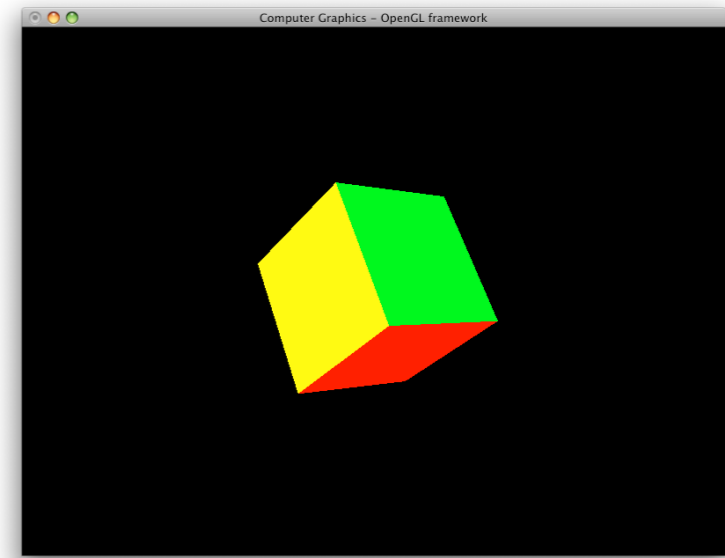
- Storing OpenGL commands for later execution
- Example, creating a display list:

```
listName = glGenLists(1);  
glNewList(listName, GL_COMPILE);  
drawSomething();  
glEndList();
```
- Calling a display list (in display function):

```
glCallList(listName);
```

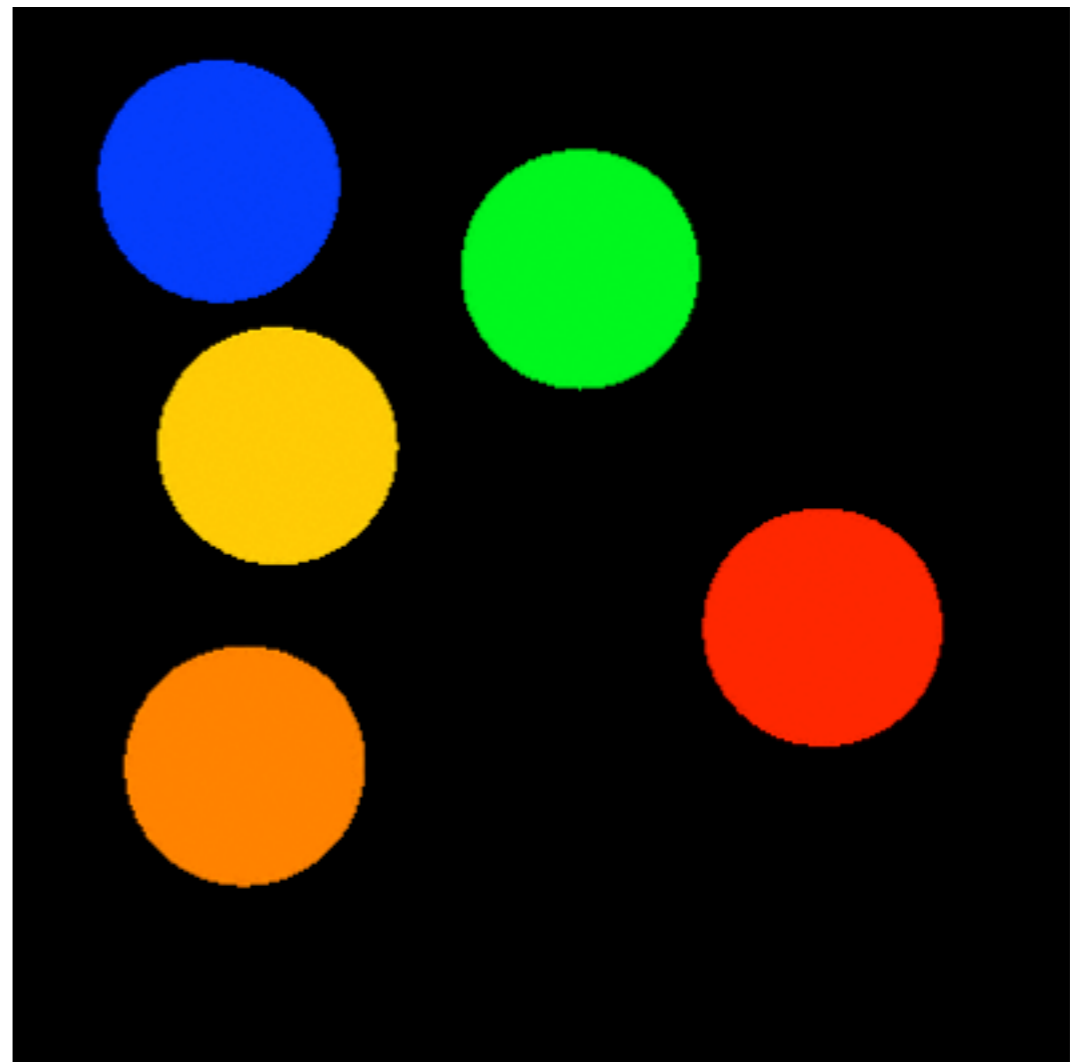
OpenGL assignments

- Add cube with each face a different color
 - Note: enable depth buffering!
- Add simple interaction capabilities (illustration)
- (Bonus) Add a better virtual trackball (link on web page)
- (Bonus) Other interaction



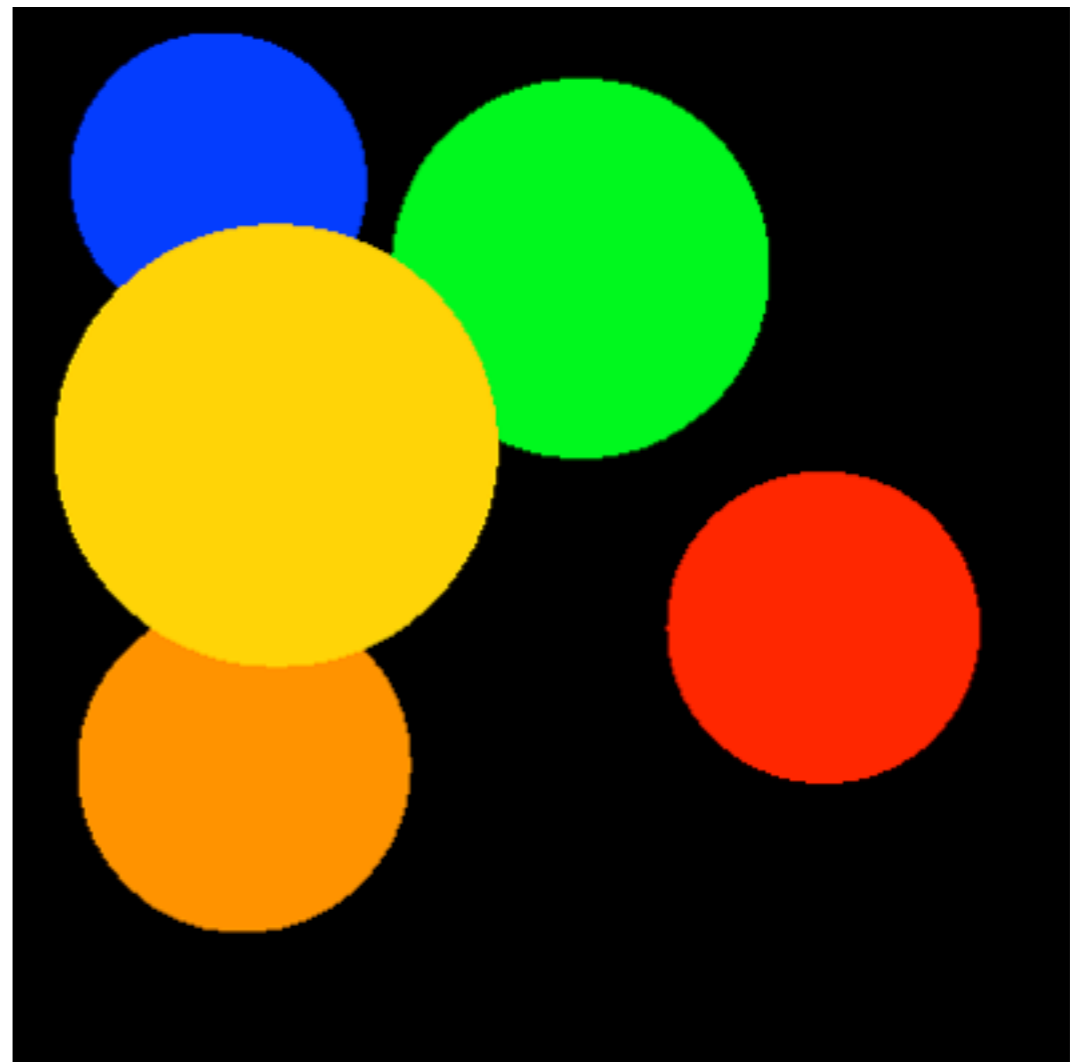
Raytracer assignment

- Implement `intersect()` function for a sphere
- Hint: remember (lookup) dot product definition



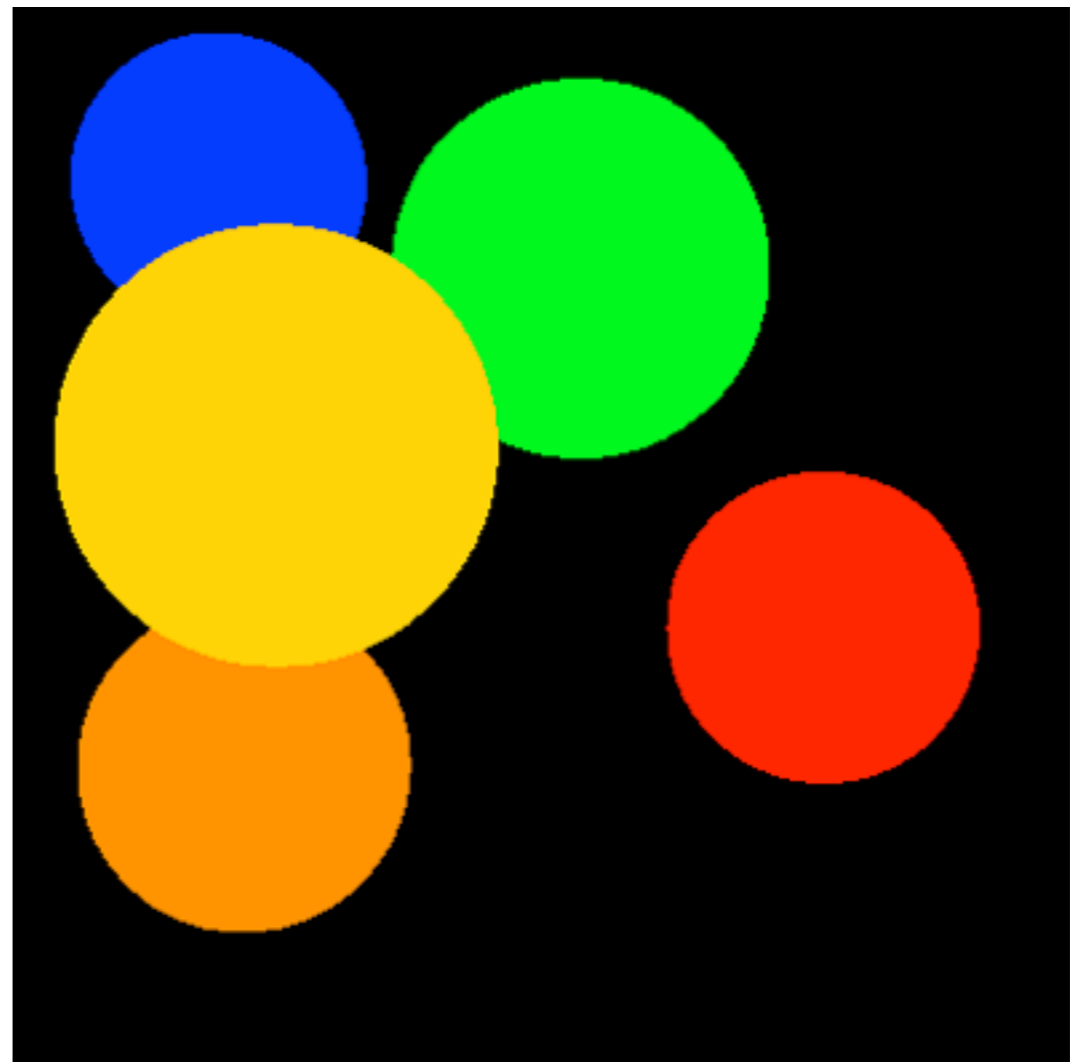
Raytracer assignment

- Implement `intersect()` function for a sphere
- Hint: remember (lookup) dot product definition



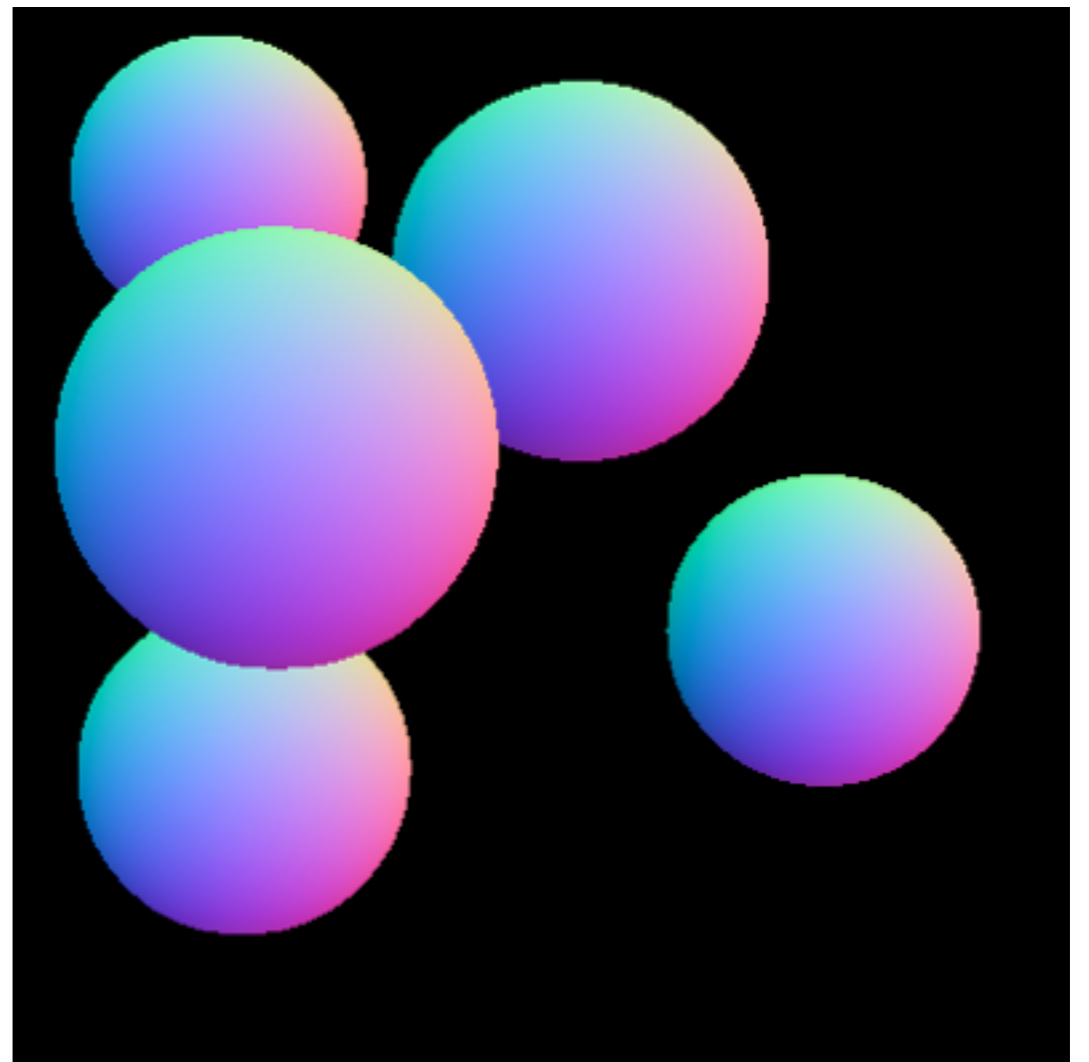
Raytracer assignment

- Implement `normal()` function for a sphere
- Produce a normal buffer image



Raytracer assignment

- Implement `normal()` function for a sphere
- Produce a normal buffer image



Raytracer assignment

- Implement `normal()` function for a sphere
- Produce a normal buffer image

