

컴퓨터그래픽스

Lab #4 - Lighting -

제 출 일	2017년 10월 31일
분 반	00반
담당교수	남병규
학 과	컴퓨터공학과
학 번	201302423
이 름	신종욱

1. Purpose of the lab

자체 발광하는 구와 빛을 반사하는 원그리기

2. Source code

```
void display()
{
    glClearColor(GL_COLOR_BUFFER_BIT);
    glMatrixMode(GL_MODELVIEW);

    glLoadIdentity();
    glPushMatrix();
    glTranslatef(0.0, 0.75, 0.0); //y축 이동
    GLfloat diffuse0[] = { 0.5, 0.0, 0.0, 1.0 };
    GLfloat ambient0[] = { 0.5, 0.0, 0.0, 1.0 };
    GLfloat specular0[] = { 0.5, 0.0, 0.0, 1.0 };
    GLfloat light0_pos[] = { 0.5, 0.0, 0.0, 1.0 };
    GLfloat shine = 120.0;
    GLfloat emission[] = { 0.5, 0.1, 0.1, 1.0 };

    glLightfv(GL_LIGHT0, GL_POSITION, light0_pos);
    glLightfv(GL_LIGHT0, GL_AMBIENT, ambient0);
    glLightfv(GL_LIGHT0, GL_DIFFUSE, diffuse0);
    glLightfv(GL_LIGHT0, GL_SPECULAR, specular0);
    glMaterialfv(GL_FRONT, GL_SHININESS, shine);
    glMaterialfv(GL_FRONT, GL_EMISSION, emission); //광원 설정
    glEnable(GL_LIGHTING);
    glEnable(GL_LIGHT0);
    glutSolidSphere(0.20, 50000, 1000);
    glPopMatrix();

    glPushMatrix();
    GLfloat ambient[] = { 0.1, 0.0, 0.0, 1.0 };
    GLfloat diffuse[] = { 0.1, 0.0, 0.0, 1.0 };
    GLfloat specular[] = { 0.1, 0.0, 0.0, 1.0 };
    glMaterialfv(GL_FRONT, GL_AMBIENT, ambient);
    glMaterialfv(GL_FRONT, GL_DIFFUSE, diffuse);
    glMaterialfv(GL_FRONT, GL_SPECULAR, specular);
    glMaterialfv(GL_FRONT, GL_SHININESS, shine); //반사될 물체 설정
    glutSolidSphere(0.20, 50000, 1000);
    glPopMatrix();
    glutSwapBuffers();
    glFlush();
}
```

붉은빛을 내는 광원을 y축으로 이동한 상태에서 구형태로 그려주고 Pop해준 후 원점에 붉은빛을 반사하는 구를 그려줬다.

3. Results

ㄱ. 광원이 없을 경우

```
void display()
{
    glClearColor(GL_COLOR_BUFFER_BIT);
    glMatrixMode(GL_MODELVIEW);

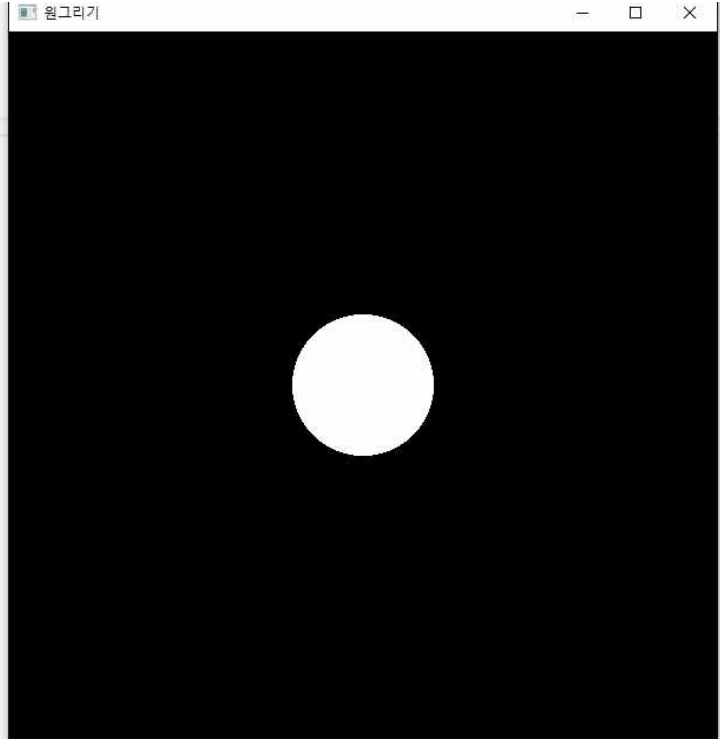
    glLoadIdentity();

    glPushMatrix();

    GLfloat ambient[] = { 0.1, 0.0, 0.0, 1.0 };
    GLfloat diffuse[] = { 0.1, 0.0, 0.0, 1.0 };
    GLfloat specular[] = { 0.1, 0.0, 0.0, 1.0 };
    GLfloat shine = 120.0;
    glMaterialfv(GL_FRONT, GL_AMBIENT, ambient);
    glMaterialfv(GL_FRONT, GL_DIFFUSE, diffuse);
    glMaterialfv(GL_FRONT, GL_SPECULAR, specular);
    glMaterialf(GL_FRONT, GL_SHININESS, shine);
    glutSolidSphere(0.20, 50000, 1000);
    glPopMatrix();
    glutSwapBuffers();
    glFlush();
}

void main(int argc, char **argv)
{
    glutInit(&argc, argv);
    glutInitDisplayMode(GLUT_DOUBLE | GLUT_RGB);
    glutInitWindowPosition(0, 0);
    glutInitWindowSize(600, 600);
    glutCreateWindow("원그리기"); //화면크기와 이름 지정
    glClearColor(0, 0, 0, 0);
    glutDisplayFunc(display); //원그리기 시작
    glutMainLoop();
}

//[CG]HW04_201302423_신종욱
```



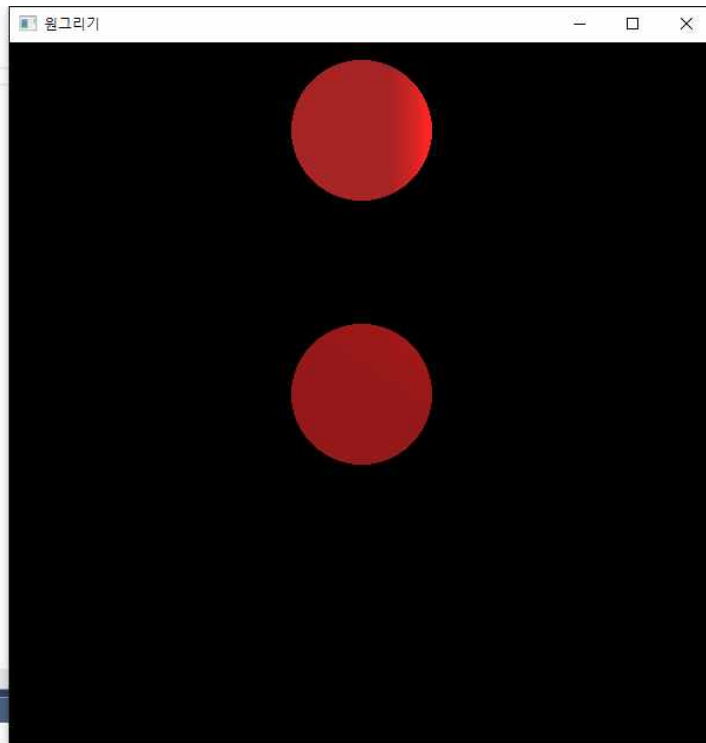
ㄴ. 광원이 붉은빛을 방출하고 물체는 붉은색을 반사하는 성질일 경우

```
glLoadIdentity();
glPushMatrix();
glTranslatef(0.0, 0.75, 0.0); //y축 이동
GLfloat diffuse0[] = { 0.5, 0.0, 0.0, 1.0 };
GLfloat ambient0[] = { 0.5, 0.0, 0.0, 1.0 };
GLfloat specular0[] = { 0.5, 0.0, 0.0, 1.0 };
GLfloat light0_pos[] = { 0.5, 0.0, 0.0, 1.0 };
GLfloat shine = 120.0;
GLfloat emission[] = { 0.5, 0.1, 0.1, 1.0 };

glLightfv(GL_LIGHT0, GL_POSITION, light0_pos);
glLightfv(GL_LIGHT0, GL_AMBIENT, ambient0);
glLightfv(GL_LIGHT0, GL_DIFFUSE, diffuse0);
glLightfv(GL_LIGHT0, GL_SPECULAR, specular0);
glMaterialf(GL_FRONT, GL_SHININESS, shine);
glMaterialfv(GL_FRONT, GL_EMISSION, emission);
glEnable(GL_LIGHTING);
glEnable(GL_LIGHT0);
glutSolidSphere(0.20, 50000, 1000);
glPopMatrix();

glPushMatrix();

GLfloat ambient[] = { 0.1, 0.0, 0.0, 1.0 };
GLfloat diffuse[] = { 0.1, 0.0, 0.0, 1.0 };
GLfloat specular[] = { 0.1, 0.0, 0.0, 1.0 };
glMaterialfv(GL_FRONT, GL_AMBIENT, ambient);
glMaterialfv(GL_FRONT, GL_DIFFUSE, diffuse);
glMaterialfv(GL_FRONT, GL_SPECULAR, specular);
glMaterialf(GL_FRONT, GL_SHININESS, shine);
glutSolidSphere(0.20, 50000, 1000);
glPopMatrix();
glutSwapBuffers();
glFlush();
```



ㄷ.광원이 다양한 색을 방출하고 하나색만 반사하는 물체일 경우

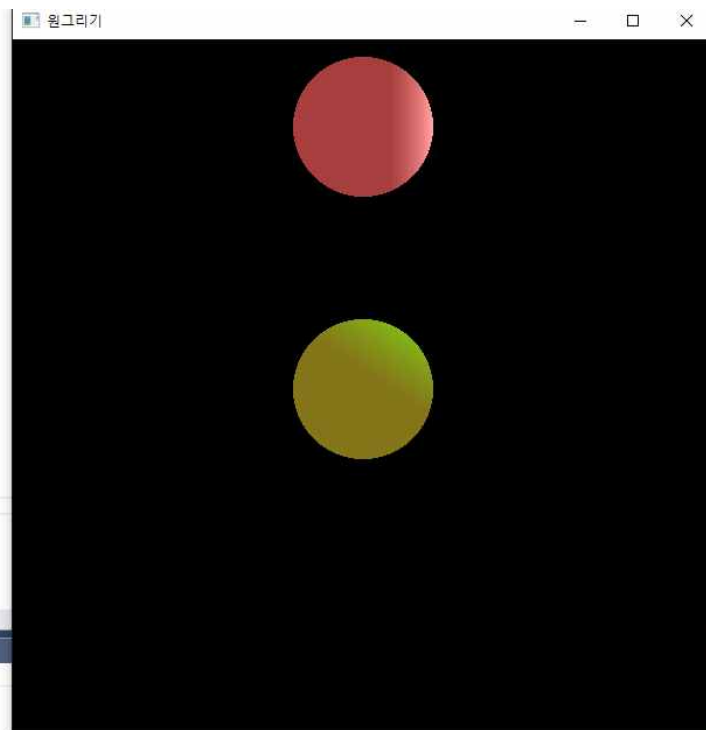
```
glLoadIdentity();
glPushMatrix();
glTranslatef(0.0, 0.75, 0.0); //y축 이동
GLfloat diffuse0[] = { 0.5, 0.5, 0.5, 1.0 };
GLfloat ambient0[] = { 0.5, 0.5, 0.5, 1.0 };
GLfloat specular0[] = { 0.5, 0.5, 0.5, 1.0 };
GLfloat light0_pos[] = { 0.5, 0.0, 0.0, 1.0 };
GLfloat shine = 120.0;
GLfloat emission[] = { 0.5, 0.1, 0.1, 1.0 };

glLightfv(GL_LIGHT0, GL_POSITION, light0_pos);
glLightfv(GL_LIGHT0, GL_AMBIENT, ambient0);
glLightfv(GL_LIGHT0, GL_DIFFUSE, diffuse0);
glLightfv(GL_LIGHT0, GL_SPECULAR, specular0);
glMaterialfv(GL_FRONT, GL_SHININESS, shine);
glMaterialfv(GL_FRONT, GL_EMISSION, emission);
glEnable(GL_LIGHTING);
glEnable(GL_LIGHT0);
glutSolidSphere(0.20, 50000, 1000);
glPopMatrix();

glPushMatrix();

GLfloat ambient[] = { 0.0, 0.5, 0.0, 1.0 };
GLfloat diffuse[] = { 0.0, 0.5, 0.0, 1.0 };
GLfloat specular[] = { 0.0, 0.5, 0.0, 1.0 };

glMaterialfv(GL_FRONT, GL_AMBIENT, ambient);
glMaterialfv(GL_FRONT, GL_DIFFUSE, diffuse);
glMaterialfv(GL_FRONT, GL_SPECULAR, specular);
glMaterialfv(GL_FRONT, GL_SHININESS, shine);
```



4. Discussions

광원이 살짝 이상하게 표현되는데 왜그런지 궁금하다.

색상선택과 반사정도를 적절히 해야지 원하는 값을 얻을수 있을거같다.