

그래픽스 중간고사(프로그램)



학과 : 컴퓨터공학과
학번 : 201611840 이름 : 이태양
제출날짜 : 2020/10/26

(1) 위 4가지 Callback 기능에 대하여 관련 자료를 찾아보세요.(그래픽스 교재나 인터넷 웹사이트에 OpenGL과 해당 키워드를 입력하면 다양한 유사 예제와 설명을 접할 수 있습니다) Open Book 이므로 과제를 수행할 때 내용들을 참고하거나 이해하면 됩니다.

sol)

Reshape Callback : 윈도우의 크기를 바꿈으로써 윈도우 내부 물체 모습 이 변화되길 기대, Reshape event 발생 간 주 : 처음 윈도우를 열 때, 윈도우 위치를 옮길 때, 윈도우 크기를 조절 할 때, 앞 윈도우에 가려져 안보이던 뒤 윈도우가 활성화 되어 앞으로 드러날 때, glutPostRedisplay() 함수에 의해 이벤트 큐에 Flag이 게시될 때.

Keyboard Callback : 문자 키나 함수 키 사용에 의한 대화형 인터페이스 설계 시 사용, Keyboard함수는 문자키만 입력받을 뿐 커서 이동키나 function key같은 특수키는 입력받을 수 없다. 특수키는 Special함수로 받는다.

Mouse Callback : 마우스 클릭 감지, 마우스 모션(motion, 움직임) 감지, 마우스가 OpenGL 창에 내부에 있는지 아니면 외부에 있는지 판단하기

Menu Callback : glutCreateMenu() : 메뉴 콜백함수를 등록하며, 그 결과 등록된 메뉴 아이디 값이 리턴 된다. 여러 가지 메뉴가 필요하면 별도로 콜백함수를 등록해야 하며 해당 메뉴 유일의 아이디 값이 리턴된다. glutSetMenu() : 현 메뉴와 파라미터에 지정한 메뉴로 설정한다. 이렇게 해서 이후 메뉴와 관련된 모든 함수는 현 메뉴에 적용된다. glutAddMenuEntry() : 현 메뉴에 메뉴 항목을 추가한다. 파라미터에는 항목의 이름과 항목이 선택될 때 메뉴 콜백 함수에게 정수 값이 전달된다. glutAddSubMenu() : 현 메뉴 항목 중 하나로서 서브메뉴 항목을 추가한다. glutAttachMenu() : 지정한 마우스 버튼에 현 메뉴를 할당한다.

(2) 콜백 기능 중 Mouse Callback을 이용하여 다음의 프로그램을 수행하시오.

① 임의의 윈도우에 1개의 3차원 물체를 display 하시오.(예: 빨간색 3D 주전자, 또는 와이어 프레임 주전자 등) 윈도우의 타이틀은 반드시 “홍길동-학번” (본인 이름과 학번을 넣으세요.)

sol)



```
#include<GL/GLUT.H>
#include<GL/GLU.H>
#include<GL/gl.h>

void my_display(void) {

    glClearColor(0, 0, 0, 0);
    glClear(GL_COLOR_BUFFER_BIT);

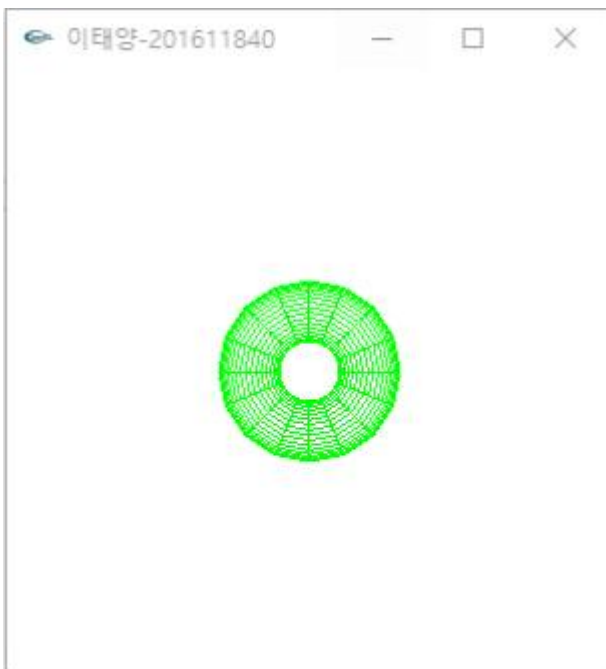
    glColor3f(4.0, 0.0, 0.0);
    glutWireTeapot(0.5);

    glFlush();
}

int main() {
    glutCreateWindow("이태양-201611840");
    glutDisplayFunc(my_display);
    glutMainLoop();
}
```

② Mouse Callback의 마우스 버튼 클릭을 통하여 기존에 display 된 물체를 다른 물체로 변경하고, 생상도 변경하는 프로그램을 작성 하고 그 결과를 제시 하시오(예: 빨간 색 3D 주전자 ---> 파란색 3D 토러스)

sol)



```

1  #include<GL/GLUT.H>
2  #include<GL/GLU.H>
3  #include<GL/gl.h>
4
5
6  GLfloat ss = 0;
7
8
9  void my_display(void) {
10
11
12      glClearColor(0, 0, 0, 0);
13      glClear(GL_COLOR_BUFFER_BIT);
14
15      glColor3f(4.0, 0.0, 0.0);
16      glutWireTeapot(0.5);
17
18
19      glFlush();
20  }
21
22  void my_change_display(void) {
23
24      glClearColor(2.0, 2.0, 2.0, 2.0);
25      glClear(GL_COLOR_BUFFER_BIT);
26
27      glColor3f(0.0, 4.0, 0.0);
28      glutWireTorus(0.1 + ss, 0.2, 35, 16);
29
30
31      glFlush();
32  }
33

```

```

void processMouse(int button, int state, int x, int y)
{
    // 왼쪽 버튼
    if (button == GLUT_LEFT_BUTTON)
    {
        my_change_display();
    }
    // 가운데 버튼
    else if (button == GLUT_MIDDLE_BUTTON)
    {
        my_change_display();
    }
    // 오른쪽 버튼
    else
    {
        my_change_display();
    }
}

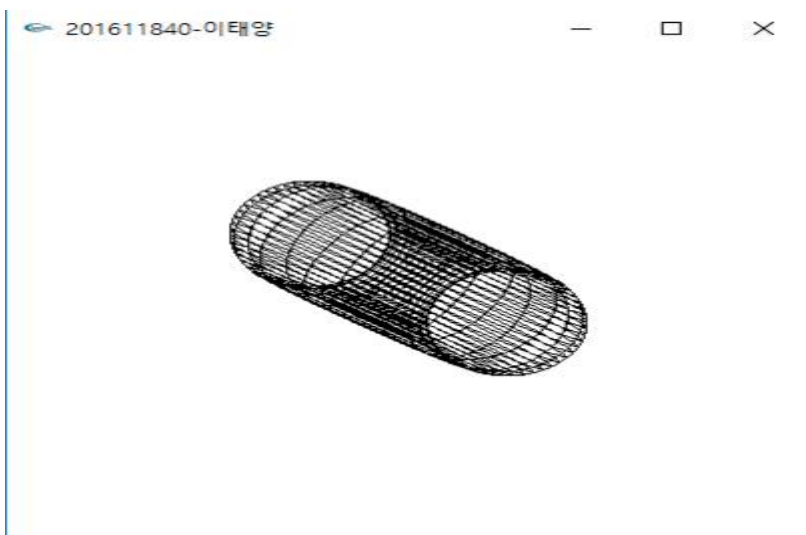
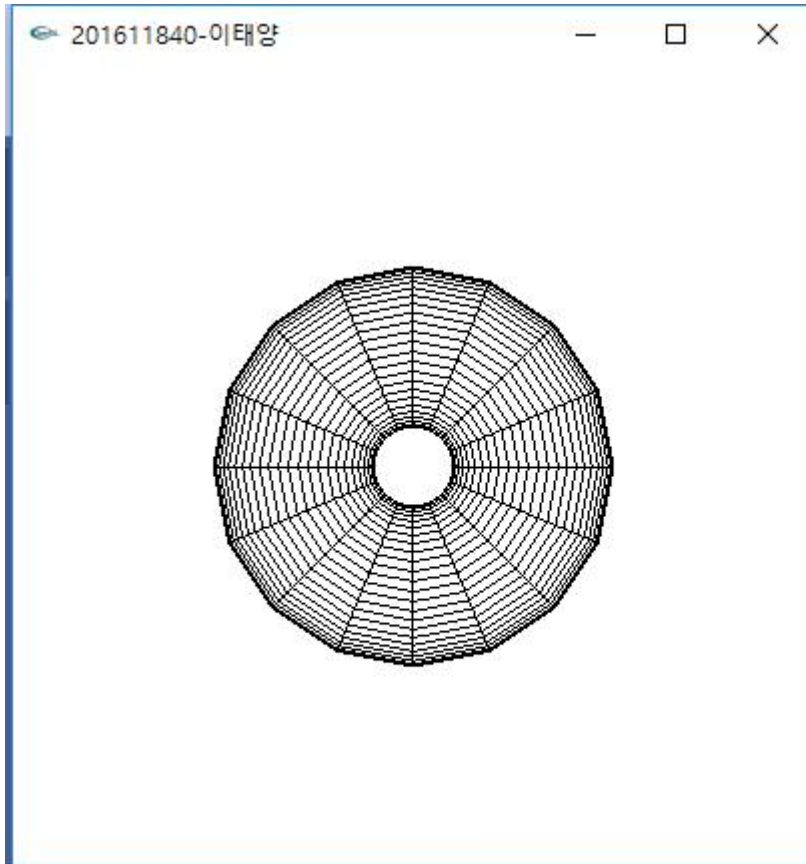
int main() {
    glutCreateWindow("이태양-201611840");

    glutDisplayFunc(my_display);
    glutMouseFunc(processMouse);
    glutMainLoop();
}

```

③ Mouse Callback의 마우스를 누른 상태로 움직여서 display 된 물체의 위치를 변경하거나 회전하는 프로그램을 작성하고 그 결과를 제시시오.

sol)



```

#include <GL/glut.h>

#include <GL/gl.h>

#include <GL/glu.h>

#include <math.h>

#include <stdio.h>

GLint xx = 0;
GLint yy = 0;
GLfloat ss = 0;
void MyDisplay() {

    glViewport(0, 0, 400, 400);

    glClear(GL_COLOR_BUFFER_BIT);

    glColor3f(0.0, 0.0, 0.0);
    glutWireTorus(0.2 + ss, 0.3, 50, 16);

    glFlush();

}

```

```

void MouseMoveEvent(GLint x, GLint y) {
    if (xx < x)

        glRotatef(2, 2, 0, 0);

    else if (xx > x)

        glRotatef(-2, 2, 0, 0);

    if (yy < y)

        glRotatef(2, 0, 2, 0);

    else if (yy > y)

        glRotatef(-2, 0, 2, 0);

    glutPostRedisplay();

    xx = x;

    yy = y;

}

```

```

void MousePressEvent(GLint Button, GLint State, GLint X, GLint Y) {
    if (Button == GLUT_DOWN) {
        glutMotionFunc(MouseMoveEvent);
    }
}

int main(int argc, char** argv) {
    glutInit(&argc, argv);

    glutInitDisplayMode(GLUT_RGB);

    glutInitWindowSize(400, 400);

    glutCreateWindow("201611840-이태양");

    glClearColor(1.0, 1.0, 1.0, 1.0);

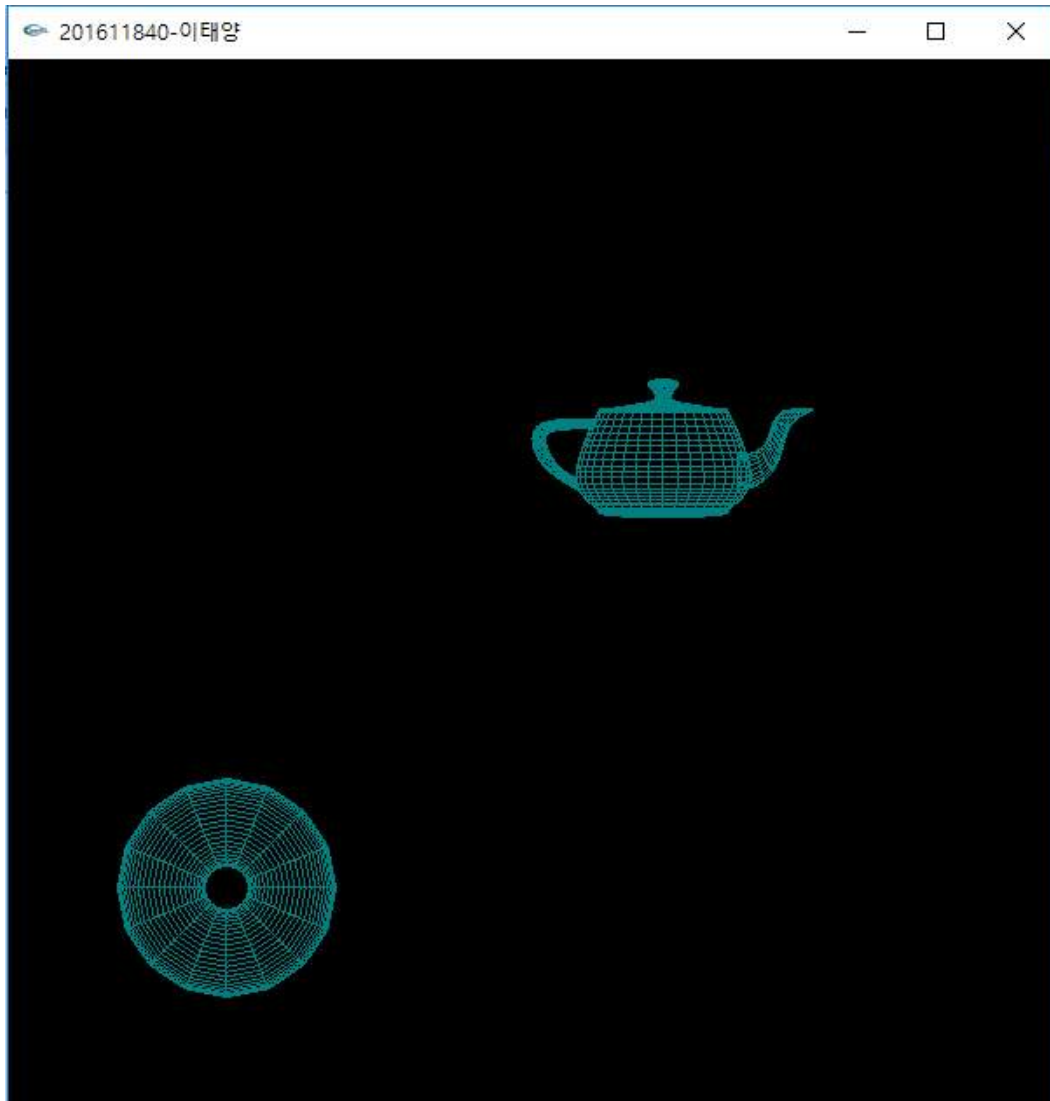
    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
    glutDisplayFunc(MyDisplay);
    glutMouseFunc(MousePressEvent);
    glutMainLoop();

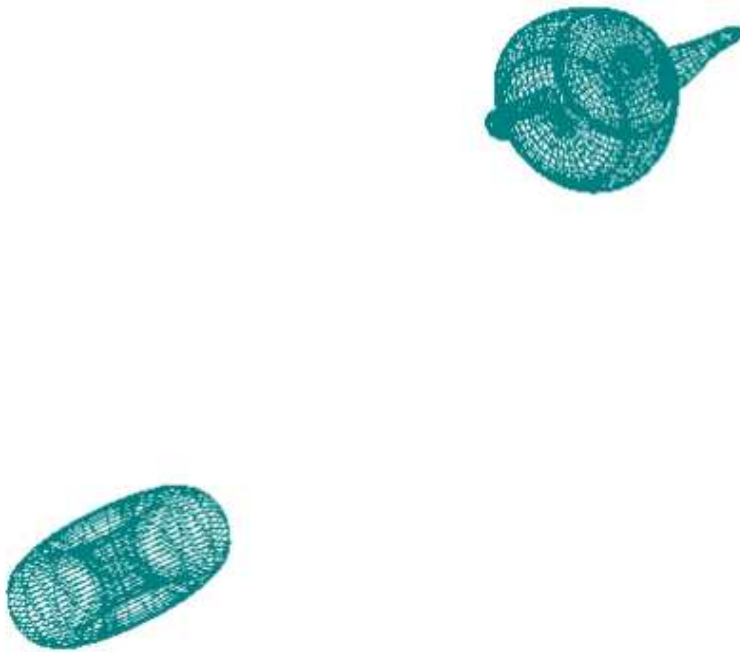
    return 0;
}

```


(3) 생성된 윈도우 안에 임의의 크기의 Viewport를 1개 이상 (1개 또는 2개) 만들고 해당 Viewport 안에서 위 (2)의 작업을 동일하게 수행하고 그 결과를 제시하시오. (윈도우에 뷰포트를 2개 만들어서 각 뷰포트 별로 Mouse Callback의 기능을 달리하는 event 기능을 만들고 제시하면 보너스 점수 부여)

sol)





```
#include <GL/glut.h>
```

```
#include <GL/gl.h>
```

```
#include <GL/glu.h>
```

```
#include <math.h>
```

```
#include <stdio.h>
```

```

GLint xx = 0;
GLint yy = 0;
GLfloat ss = 0;
int Width, Height;
void MyDisplay() {

    glClear(GL_COLOR_BUFFER_BIT);

    glClearColor(1, 1, 1, 0);
    glColor3f(0.0, 0.5, 0.5);
    glViewport(0, 0, Width / 2, Height / 2);
    glutWireTorus(0.2 + ss, 0.3, 50, 16);
    glViewport(Width/2, Height/2, Width / 2, Height / 2);
    glutWireTeapot(0.4 + ss);
    glFlush();

}

void MouseMoveEvent(GLint x, GLint y) {
    if (xx < x)

        glRotatef(2, 2, 0, 0);

    else if (xx > x)

        glRotatef(-2, 2, 0, 0);

    if (yy < y)

        glRotatef(2, 0, 2, 0);

    else if (yy > y)

        glRotatef(-2, 0, 2, 0);

    glutPostRedisplay();

    xx = x;

```

```

        yy = y;

    }

void MousePressEvent(GLint Button, GLint State, GLint X, GLint Y) {

    if (Button == GLUT_DOWN) {

        glutMotionFunc(MouseMoveEvent);

    }

}

int main(int argc, char** argv) {
    Width = 500; Height = 500;
    glutInit(&argc, argv);

    glutInitDisplayMode(GLUT_RGB);

    glutInitWindowSize(600, 600);

    glutCreateWindow("201611840-이태양");

    glClearColor(0.0, 0.0, 0.0, 0.0);

    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
    glutDisplayFunc(MyDisplay);
    glutMouseFunc(MousePressEvent);
    glutMainLoop();

    return 0;

}

```