LAB SET 2

Write a program to demonstrate the two views, orthographic view and perspective view, of OpenGL by consider a 2D and a 3D with any two primitives.

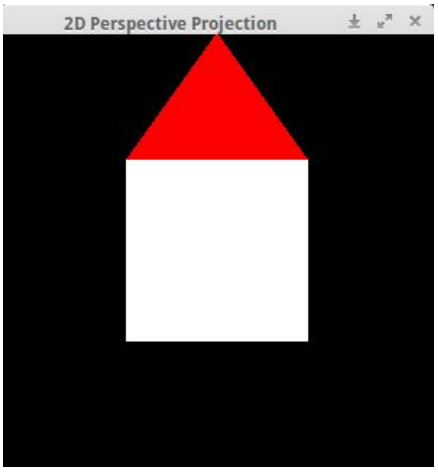
```
//Akarsh Singh  1SI16CS007  LABSET 2
#include <stdio.h>
#include<GL/glut.h>

void display2D()
{
     glClear(GL_COLOR_BUFFER_BIT);
     glColor3f(1.0,0.0,0.0);
     glBegin(GL_QUADS);
     glColor3f(1.0,1.0,1.0);
     glVertex2d(-0.5,-0.5);
     glVertex2d(0.5,-0.5);
     glVertex2d(0.5,0.5);
     glVertex2d(-0.5,0.5);
     glEnd();
     glBegin(GL_TRIANGLES);
     glColor3f(1.0,0.0,0.0);
     glVertex2d(0.5,0.5);
     glVertex2d(-0.5,0.5);
     glVertex2d(0.0,1.2);
     glEnd();
     glFlush();
}

void display3D()
{
     glClear(GL_COLOR_BUFFER_BIT|GL_DEPTH_BUFFER_BIT);
     glClearColor(0.0,0.0,0.0,0.0);
     glBegin(GL_LINE_LOOP);
     glVertex2f(0.5,0.5);
     glVertex2f(-0.5,0.5);
     glVertex2f(-0.5,-0.5);
     glVertex2f(0.5,-0.5);
     glEnd();
```

```
     glBegin(GL_LINE_LOOP);
     glVertex2f(0.7,0.7);
     glVertex2f(-0.3,0.7);
     glVertex2f(-0.3,-0.3);
     glVertex2f(0.7,-0.3);
     glEnd();
     glBegin(GL_LINES);
     glVertex2f(0.5,0.5);
     glVertex2f(0.7,0.7);
     glVertex2f(-0.3,0.7);
     glVertex2f(-0.5,0.5);
     glVertex2f(-0.3,-0.3);
     glVertex2f(-0.5,-0.5);
     glVertex2f(0.7,-0.3);
     glVertex2f(0.5,-0.5);
     glEnd();
     glFlush();
}

void init2D(int ch)
{
     glutInitDisplayMode(GLUT_SINGLE|GLUT_RGB);

     if(ch==1)
          glutCreateWindow("2D Orthogonal Projection");
     else
          glutCreateWindow("2D Perspective Projection");

     glutDisplayFunc(display2D);
     glMatrixMode(GL_PROJECTION);
     glLoadIdentity();
     if(ch==1)
          gluOrtho2D(-2.0,2.0,-2.0,2.0);
     else
          gluPerspective(100.0,1.0,1.0,100.0);

     gluLookAt(0.0,0.0,1.0,0.0,0.0,0.0,0.0,1.0,0.0);
     glMatrixMode(GL_MODELVIEW);
     glLoadIdentity();
}

void init3D(int ch)
```

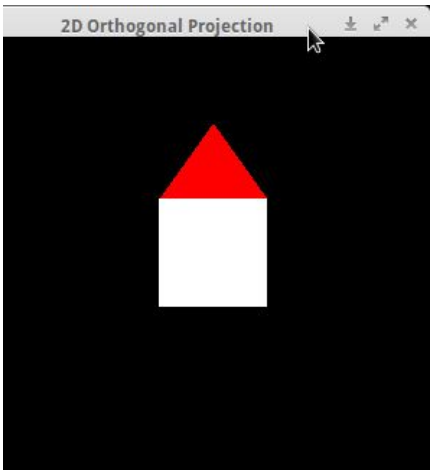
```
{
     glutInitDisplayMode(GLUT_SINGLE|GLUT_RGB|GLUT_DEPTH);
     if(ch==1)
          glutCreateWindow("3D Orthogonal Projection");
     else
          glutCreateWindow("3D Perspective Projection");

     glutDisplayFunc(display3D);
     glMatrixMode(GL_PROJECTION);
     glLoadIdentity();
     if(ch==1)
          glOrtho(-2.0,2.0,-2.0,2.0,-2.0,2.0);
     else
          glFrustum(-2.0,2.0,-2.0,2.0,-2.0,2.0);

     glMatrixMode(GL_MODELVIEW);
     glLoadIdentity();
     glEnable(GL_DEPTH_TEST);
}

int main(int argc, char **argv)
{
     int ch;
     glutInit(&argc,argv);
     printf("Enter the choice: \n 1: 2D Orthogoal Projection
     \n2: 2D Perspective Projection\n 3: 3D orthogoal
     Projection\n 4: 3D Perspective Projection\n");
     scanf("%d", &ch);
     switch(ch)
     {
          case 1:   init2D(1);
                    break;
          case 2:   init2D(2);
                    break;
          case 3:   init3D(1);
                    break;
          case 4:   init3D(2);
                    break;
     }
     glutMainLoop();
     return 0;
}
```

OUTPUT

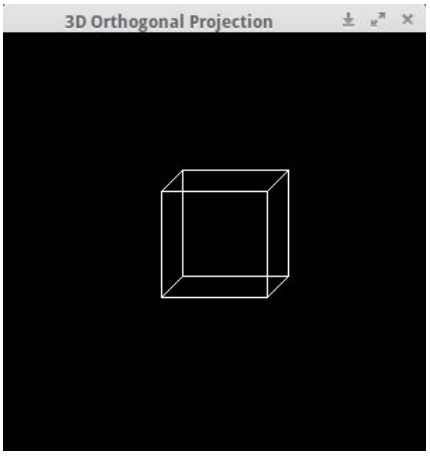
2D Perspective Projection



2D Orthogonal Projection



3D Orthogonal Projection



3D Perspective Projection