



Daffodil
International
University

LAB-REPORT

Course Title : Computer graphics Lab.

Course Code : CSE-422(lab).

Experiment No: 02.

Experiment Name: Draw the below diagram in code blocks and Opengl .

Submitted To :

Name : Md.Atik Asif Khan Akash (LECTURER).

Department of : CSE.

Daffodil International University.

Submitted By :

Name: MD. Hasan Ahmad.

ID: 201-15-3160.

Department of : B.sc in CSE.

Section : PC-D.

Date of submission: 27/01 / 2023.

Draw the below diagram in code blocks and Opengl .

Code:

```
#include<windows.h>
#include <GL/gl.h>
#include <GL/glut.h>
void display(void)
{
    /* clear all pixels */
    glClear (GL_COLOR_BUFFER_BIT);
    /* draw white polygon (rectangle) with corners at
    * (0.25, 0.25, 0.0) and (0.75, 0.75, 0.0)
    */

    // First Square but here we will code it Quads
    glBegin(GL_QUADS); //Begin quadrilateral coordinates
    glColor3f (1.0,0.0,0.0);
        //Quads
        glVertex3f(0.05f, 0.65f, 0.0f);
        glVertex3f(0.35f, 0.65f, 0.0f);
        glVertex3f(0.35f, 0.85f, 0.0f);
        glVertex3f(0.05f, 0.85f, 0.0f);

    glEnd(); //End quadrilateral coordinates

    //seconds Square but here we will code it Quads
    glBegin(GL_QUADS); //Begin quadrilateral coordinates
    glColor3f (0.0,1.0,0.0);
        //Quads
        glVertex3f(0.05f, 0.35f, 0.0f);
        glVertex3f(0.35f, 0.35f, 0.0f);
        glVertex3f(0.35f, 0.55f, 0.0f);
        glVertex3f(0.05f, 0.55f, 0.0f);

    glEnd(); //End quadrilateral coordinates

    //Third Square but here we will code it Quads
    glBegin(GL_QUADS); //Begin quadrilateral coordinates
    //glColor3f (1.0,1.0,1.0);
    //glColor3f (0.0,0.0,0.0);
    //Quads
        glVertex3f(0.0f, 0.30f, 0.0f);
        //glColor3f (0.20,0.20,0.20);
        glColor3f (0.80,0.80,0.80);

        glVertex3f(0.20f, 0.305f, 0.0f);
        glColor3f (0.20,0.20,0.20);
```

```

glVertex3f(0.20f, 0.45f, 0.0f);
glColor3f (0.80,0.80,0.80);

glVertex3f(0.00f, 0.45f, 0.0f);
glColor3f (0.20,0.20,0.20);

glEnd(); //End quadrilateral coordinates

//Triangle
glBegin(GL_TRIANGLES);
glColor3f (0.0,0.0,1.0);

glVertex3f(0.55f, 0.35f, 0.0f);
glVertex3f(0.80f, 0.35f, 0.0f);
glVertex3f(0.67f, 0.55f, 0.0f);
glEnd();//End triangle coordinates

//Triangle
glBegin(GL_TRIANGLES);
//glColor3f (0.0,1.0,0.0);

glVertex3f(0.65f, 0.45f, 0.0f);
glColor3f (1.0,0.0,0.0);

glVertex3f(0.90f, 0.45f, 0.0f);
glColor3f (0.0,1.0,.0);

glVertex3f(0.77f, 0.25f, 0.0f);
glColor3f (0.0,0.0,1.0);
glEnd();//End triangle coordinates

// polygon
glBegin(GL_POLYGON); //Begin polygoncoordinates
glColor3f (1.0,1.0,0.0);
//polygon
glVertex3f(0.5f, 0.63f, 0.0f);
glVertex3f(0.6f, 0.63f, 0.0f);
glVertex3f(0.7f, 0.73f, 0.0f);
glVertex3f(0.6f, 0.83f, 0.0f);
glVertex3f(0.5f, 0.83f, 0.0f);
glVertex3f(0.4f, 0.73f, 0.0f);

glEnd(); //End polygon coordinates

```

```

/* don't wait!
* start processing buffered OpenGL routines
*/
glFlush ();
}
void init (void)
{
/* select clearing (background) color */
glClearColor (0.0, 0.0, 0.0, 0.0);
/* initialize viewing values */
glMatrixMode(GL_PROJECTION);
glLoadIdentity();
glOrtho(0.0, 1.0, 0.0, 1.0, -10.0, 10.0);
}
/*
* Declare initial window size, position, and display mode
* (single buffer and RGBA). Open window with "hello"
* in its title bar. Call initialization routines.
* Register callback function to display graphics.
* Enter main loop and process events.
*/
int main(int argc, char** argv)
{
glutInit(&argc, argv);
glutInitDisplayMode (GLUT_SINGLE | GLUT_RGB);
glutInitWindowSize (600, 600);
glutInitWindowPosition (100, 100);
glutCreateWindow ("hello");
init ();
glutDisplayFunc(display);
glutMainLoop();
return 0; /* ISO C requires main to return int. */
}

```

ScreenShot:

