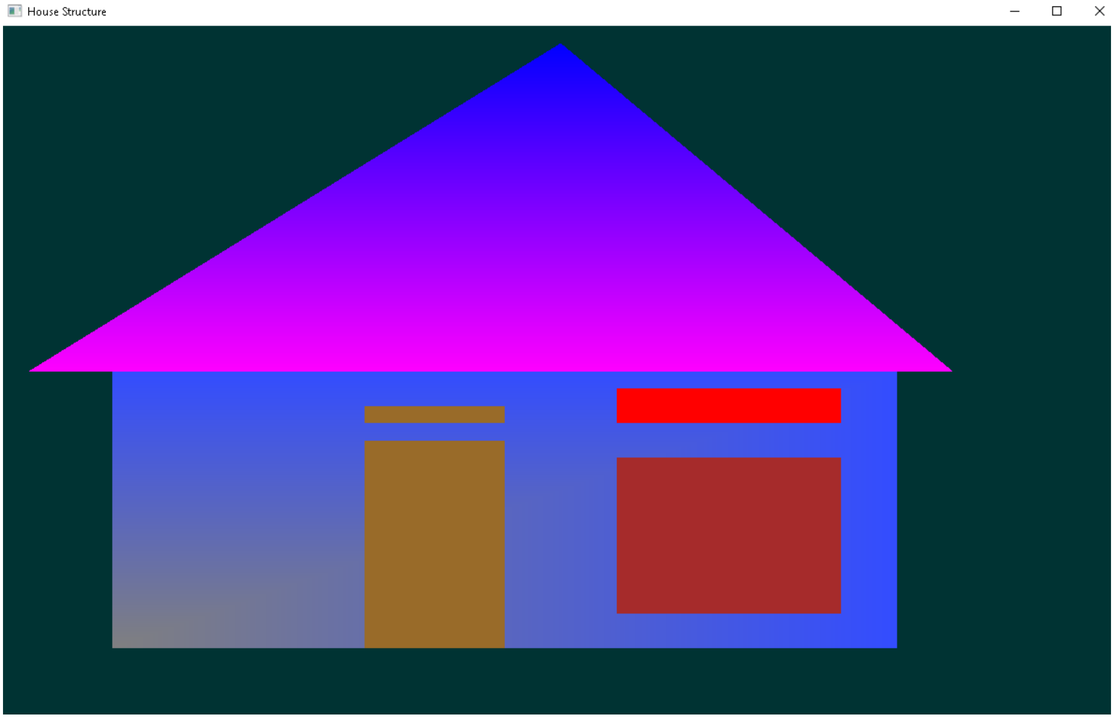
ASSIGNMENT -6
COMPUTER GRAPHICS

– Shubham Tiwari
101916126
3CS12

1. Basic House Like Structure

```
#include<GL\glut.h>
#include<iostream>
#include<windows.h>

using namespace std;

void myInit()
{
	glClearColor(0, 0.2, 0.2, .5);
	//glClear(GL_COLOR_BUFFER_BIT);
	glColor3f(1.0, 0.3, 0.9);
	glMatrixMode(GL_PROJECTION);
	glLoadIdentity();
	gluOrtho2D(0.0, 400.0, 0.0, 400.0);
}


void Polygon()
{
	// Hut Body
	glClear(GL_COLOR_BUFFER_BIT);
	glPointSize(4.0);
	glColor3f(0.5f, 0.5f, 0.5f);
	glBegin(GL_POLYGON);
	glVertex2i(40, 40);
	glVertex2i(320, 40);
	glVertex2i(40, 200);
	glColor3f(0.2f, 0.3f, 1.0f);
	glVertex2i(320, 200);
	glVertex2i(40, 200);
	glVertex2i(40, 40);
	glVertex2i(320, 200);
	glVertex2i(320, 40);
	glEnd();
	// Hut Window
	glColor3f(0.65f, 0.17f, 0.17f);
	glBegin(GL_POLYGON);
	glVertex2i(220, 60);
	glVertex2i(300, 60);
	glVertex2i(220, 150);
	glVertex2i(300, 150);
	glVertex2i(220, 60);
	glVertex2i(220, 150);
	glVertex2i(300,150);
	glVertex2i(300, 60);
	glEnd();
	// Hut Window Roof
	glColor3f(1.0f, 0.0f, 0.0f);
	glBegin(GL_POLYGON);
	glVertex2i(220, 170);
	glVertex2i(300, 170);
	glVertex2i(220, 190);
	glVertex2i(300, 190);
	glVertex2i(220, 170);
	glVertex2i(220, 190);
	glVertex2i(300,190);
	glVertex2i(300, 170);
	glEnd();
	// Hut Door
	glColor3f(0.60f, 0.42f, 0.16f);
	glBegin(GL_POLYGON);
	glVertex2i(130, 40);
	glVertex2i(130, 160);
	glVertex2i(130,160);
	glVertex2i(180, 160);
	glVertex2i(180,100);
	glVertex2i(180, 40);
	glVertex2i(120, 40);
	glVertex2i(170, 40);
	glEnd();
	// Hut Door Roof
	glColor3f(0.60f, 0.42f, 0.16f);
	glBegin(GL_POLYGON);
	glVertex2i(130, 170);
	glVertex2i(130, 180);
	glVertex2i(130,180);
	glVertex2i(180, 180);
	glVertex2i(180,170);
	glVertex2i(180, 180);
	glVertex2i(130,170);
	glVertex2i(180, 170);
	glEnd();
	// Roof Triangular
	glColor3f(1.0f, 0.0f, 1.0f);
	glBegin(GL_POLYGON);
	glVertex2i(10, 200);
	glVertex2i(340, 200);
	glColor3f(0.0f, 0.0f, 1.0f);
	glVertex2i(200, 390);
	glVertex2i(10, 200);
	glVertex2i(200, 390);
	glEnd();
	glFlush();


}


void myDisplay()
{
	Polygon();
	glFlush();
}

int main(int argc, char** argv)
{

	glutInit(&argc, argv);
	glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
	glutInitWindowSize(1200, 740);
```

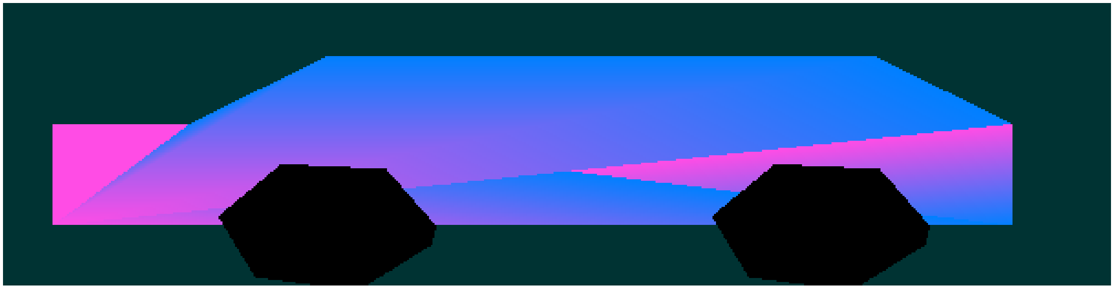
```
        glutInitWindowPosition(0, 0);
        glutCreateWindow("House Structure");
        glutDisplayFunc(myDisplay);
        myInit();
        glutMainLoop();
        return 0;
}
```



2. Car Structure

```
#include<GL\glut.h>
#include<iostream>
#include<windows.h>

using namespace std;

void myInit()
{
        glClearColor(0, 0.2, 0.2, .5);
        glColor3f(1.0, 0.3, 0.9);
        glMatrixMode(GL_PROJECTION);
        glLoadIdentity();
        gluOrtho2D(0.0, 400.0, 0.0, 400.0);
}
void wheel(int x, int y)
{
        float th;
        glBegin(GL_POLYGON);
        glColor3f(0, 0, 0);
        for (int i = 0; i < 360; i++)
        {
                int th = i * (3.1416 / 180); glVertex2f(x + 20 * cos(th), y +
20 * sin(th));
        }
        glEnd();
}
```

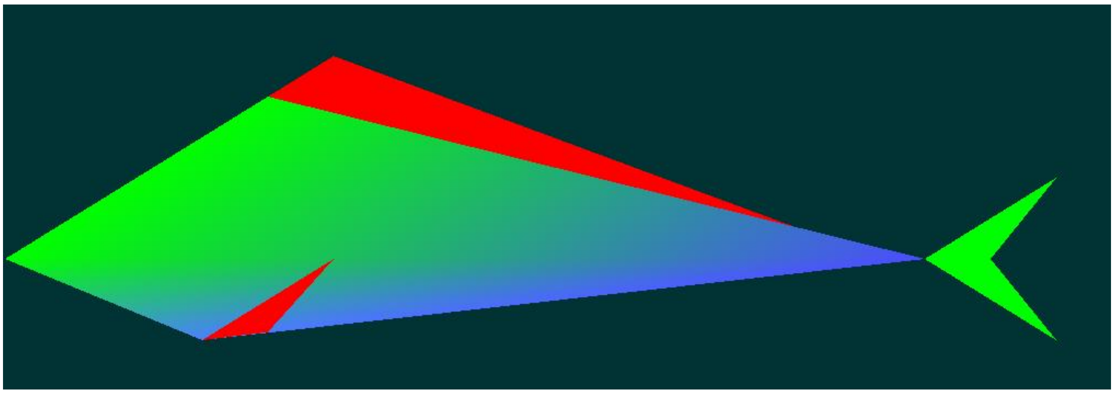
```
void Polygon()
{
	// Car Body
	glClear(GL_COLOR_BUFFER_BIT);
	glBegin(GL_POLYGON);
	glVertex2f(50, 50);
	glVertex2f(50, 80);
	glVertex2f(225, 80);
	glColor3f(0.0f, 0.5f, 1.0f);
	glVertex2f(225, 50);

	glBegin(GL_POLYGON);
	glVertex2f(75, 80);
	glColor3f(0.0f, 0.5f, 1.0f);
	glVertex2f(100, 100);
	glVertex2f(200, 100);
	glVertex2f(225, 80);
	glEnd();
	wheel(100, 50);
	wheel(190, 50);
	glFlush();


}


void myDisplay()
{
	Polygon();
	glFlush();
}

int main(int argc, char** argv)
{

	glutInit(&argc, argv);
	glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
	glutInitWindowSize(1200, 740);
	glutInitWindowPosition(0, 0);
	glutCreateWindow("Car Structure");
	glutDisplayFunc(myDisplay);
	myInit();
	glutMainLoop();
	return 0;
}
```



3. Fish Structure

```
#include<GL\glut.h>
#include<iostream>
#include<windows.h>
```

```
using namespace std;

void myInit()
{
	glClearColor(0, 0.2, 0.2, .5);
	glColor3f(1.0, 0.3, 0.9);
	glMatrixMode(GL_PROJECTION);
	glLoadIdentity();
	gluOrtho2D(0.0, 400.0, 0.0, 400.0);
}

void Polygon()
{
	glClear(GL_COLOR_BUFFER_BIT);
	glBegin(GL_POLYGON);
	glColor3f(0.0, 1.0, 0.0);
	glVertex2i(40, 200);
	glVertex2i(120, 280);
	glColor3f(0.3, 0.3, 1.0);
	glVertex2i(320, 200);
	glColor3f(0.3, 0.5, 1.0);
	glVertex2i(100, 160);
	glEnd();
	glPointSize(20);
	glBegin(GL_POINT);
	glColor3f(0.0, 0.0, 0.0);
	glVertex2i(60, 200);
	glEnd();

	glBegin(GL_POLYGON);
	glColor3f(0.0, 1.0, 0.0);
	glVertex2i(320, 200);
	glVertex2i(360, 240);
	glVertex2i(340, 200);
	glVertex2i(360, 160);
	glColor3f(0.0, 0.5, 0.5);
	glVertex2i(320, 200);
	glEnd();
	glBegin(GL_POLYGON);
	glColor3f(1.0, 0.0, 0.0);
	glVertex2i(120, 280);
	glVertex2i(140, 300);
	glVertex2i(280, 216);
	glColor3f(0.0, 0.5, 0.5);
	glVertex2i(120, 280);
	glEnd();
	glBegin(GL_POLYGON);
	glColor3f(1.0, 0.0, 0.0);
	glVertex2i(100, 160);
	glVertex2i(140, 200);
	glVertex2i(120, 164);
	glColor3f(0.0, 0.5, 0.5);
	glVertex2i(100, 160);
	glEnd();

	glFlush();


}


void myDisplay()
{
        Polygon();
        glFlush();
}

int main(int argc, char** argv)
{

        glutInit(&argc, argv);
        glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
        glutInitWindowSize(1200, 740);
        glutInitWindowPosition(0, 0);
        glutCreateWindow("Fish Art");
        glutDisplayFunc(myDisplay);
        myInit();
        glutMainLoop();
        return 0;
}
```



4. Man Art

```
#include<GL\glut.h>
#include<iostream>
#include<windows.h>

using namespace std;

void myInit()
{
        glClearColor(0, 0.0, 0.2, 1.0);
        glColor3f(1.0, 0.3, 0.9);
        glMatrixMode(GL_PROJECTION);
        glLoadIdentity();
        gluOrtho2D(0.0, 400.0, 0.0, 400.0);
}
void wheel(int x, int y)
{
        float th;
        glBegin(GL_POLYGON);
        glColor3f(0.3, 0.7, 0.4);
        for (int i = 0; i < 360; i++)
        {
                th = i * (3.1416 / 180);
                glVertex2f(x + 30 * cos(th), y + 30 * sin(th));
                glColor3f(i / 360.0, 0.7, 0.4);
        }
        glEnd();
}
void Polygon()
{
	glClear(GL_COLOR_BUFFER_BIT);
	glColor3f(0.0, 100.0, 0.0);
	glBegin(GL_LINES);
	//body
	glVertex2f(250.0, 300.0);
	glVertex2f(250.0, 100.0);
	//legs
	glVertex2f(250.0, 100.0);
	glVertex2f(150.0, 40.0);
	glVertex2f(250.0, 100.0);
	glVertex2f(350.0, 40.0);
	//arms
	glVertex2f(250.0,180.0);
	glVertex2f(150.0,210.0);
	glVertex2f(250.0,180.0);
	glVertex2f(350.0,210.0);
	glEnd();
	wheel(250, 300);

	glFlush();

}


void myDisplay()
{
	Polygon();
	glFlush();
}

int main(int argc, char** argv)
{

	glutInit(&argc, argv);
	glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
	glutInitWindowSize(1200, 740);
	glutInitWindowPosition(0, 0);
	glutCreateWindow("Man Art");
	glutDisplayFunc(myDisplay);
	myInit();
	glutMainLoop();
	return 0;
}
```