



**NOTRE DAME UNIVERSITY**  
**BANGLADESH**

**Computer Graphics Lab Report-01**

**Course Code: CSE-4204**

**Course Title: Computer Graphics Lab**

**Lab Task Topic: Setting Up Glut & Draw a House**

**Submitted by:**

**Name: Istiak Alam**

**ID: 0692230005101005**

**Batch: CSE-20**

**Submission Date: January 17, 2026**

**Submitted to:**

**Humayara Binte Rashid**

**Lecturer, Dept. of CSE**

**Notre Dame University Bangladesh**

# Table of Contents

<b>1</b>	<b>Objective</b>	<b>1</b>
<b>2</b>	<b>Tools and Environment</b>	<b>1</b>
<b>3</b>	<b>Graph Implementation</b>	<b>1</b>
<b>4</b>	<b>Source Code</b>	<b>2</b>
<b>5</b>	<b>Output</b>	<b>4</b>
<b>6</b>	<b>Discussion</b>	<b>4</b>
<b>7</b>	<b>Conclusion</b>	<b>4</b>

# 1 Objective

The objective of this lab is to understand the basics of Computer Graphics programming using OpenGL and GLUT. In this experiment, a simple house structure is created using basic geometric primitives such as quadrilaterals and triangles in a 2D orthographic projection.

## 2 Tools and Environment

- Programming Language: C++
- Graphics Library: OpenGL with GLUT
- IDE: Code::Blocks
- Operating System: Kali Linux

## 3 Graph Implementation

Here is the Graph file [Link](#)

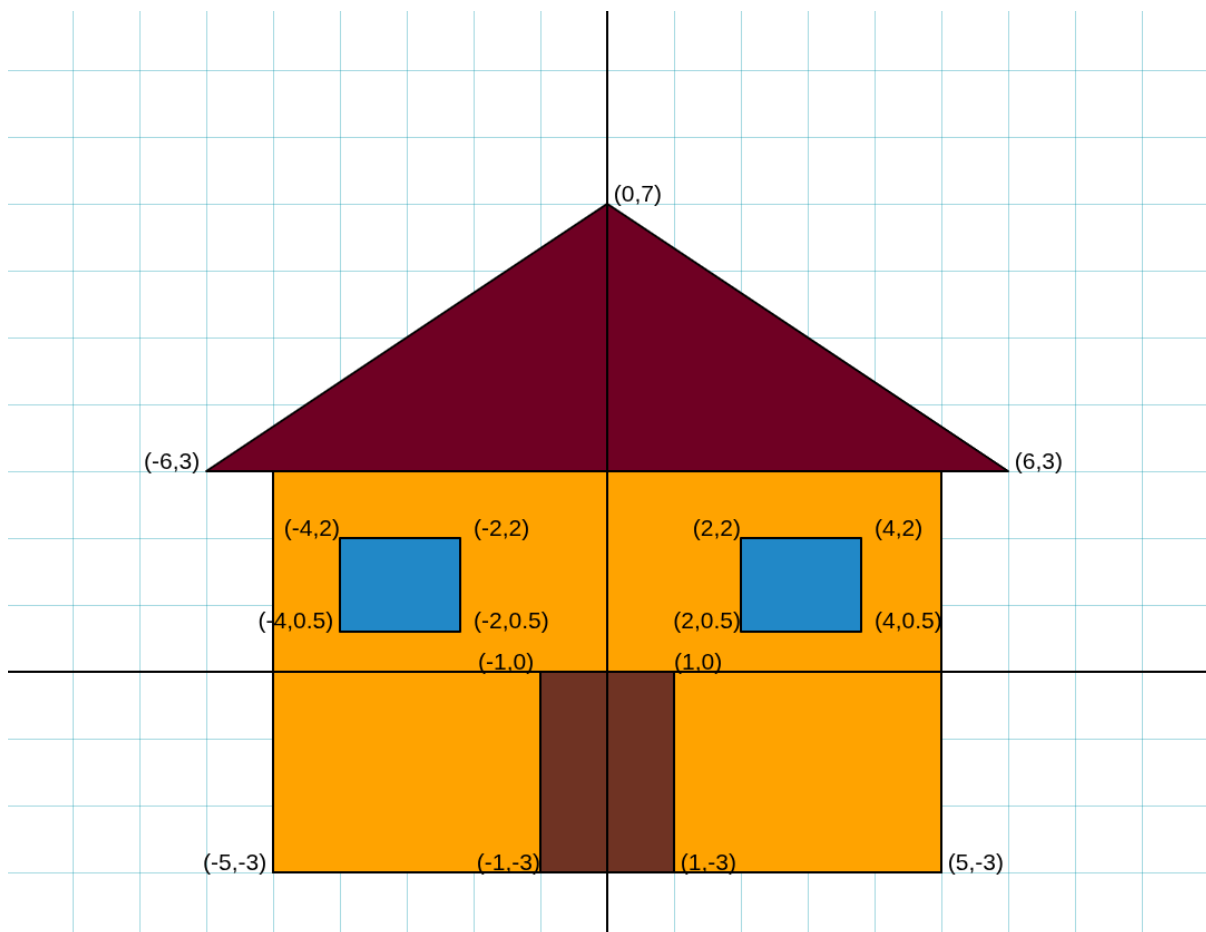


Figure 1: Graph View

## 4 Source Code

The program begins by initializing the OpenGL environment. The `init()` function sets the background color to white and defines a 2D orthographic viewing volume. The `display()` function is responsible for rendering the house. Different parts of the house such as the base, roof, door, and windows are drawn using OpenGL primitives. Colors are assigned using the `glColor3f()` function. The main function initializes GLUT, creates a window, registers the display callback, and starts the event loop.

```
/*
 * Lab Task-01
 * Creating a House using quad line
 * Written by Istiak Alam
 */

#include <GL/glut.h> // GLUT header is important
#include <stdlib.h>
#include <math.h>
#include <cmath>

// Setup and Projection
void init() {
    glClearColor(1.0f, 1.0f, 1.0f, 1.0f); // White background
    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
    glOrtho(-10, 10, -10, 10, -1, 1); // Its just like margine
    glMatrixMode(GL_MODELVIEW);
}

void display() { // Display function
    glClear(GL_COLOR_BUFFER_BIT);

    glColor3f(1.0f, 0.8f, 0.0f); // Light orange
    glBegin(GL_QUADS); // House Base (Rectangle)
        glVertex2f(-5.0f, -3.0f);
        glVertex2f( 5.0f, -3.0f);
        glVertex2f( 5.0f,  3.0f);
        glVertex2f(-5.0f,  3.0f);
    glEnd();

    glColor3f(0.8f, 0.0f, 0.0f); // Dark red
    glBegin(GL_TRIANGLES); // Roof (Triangle)
        glVertex2f(-6.0f,  3.0f);
        glVertex2f( 6.0f,  3.0f);
        glVertex2f( 0.0f,  7.0f);
    glEnd();

    glColor3f(0.3f, 0.2f, 0.1f); // Brown
    glBegin(GL_QUADS); // Door (Rectangle)
```

```
        glVertex2f(-1.0f, -3.0f);
        glVertex2f( 1.0f, -3.0f);
        glVertex2f( 1.0f,  0.0f);
        glVertex2f(-1.0f,  0.0f);
    glEnd();

    glColor3f(0.0f, 0.6f, 1.0f); // Light blue
    glBegin(GL_QUADS);           // Window 1 (Left)
        glVertex2f(-4.0f,  0.5f);
        glVertex2f(-2.0f,  0.5f);
        glVertex2f(-2.0f,  2.0f);
        glVertex2f(-4.0f,  2.0f);
    glEnd();

    glBegin(GL_QUADS);           // Window 2 (Right)
        glVertex2f( 2.0f,  0.5f);
        glVertex2f( 4.0f,  0.5f);
        glVertex2f( 4.0f,  2.0f);
        glVertex2f( 2.0f,  2.0f);
    glEnd();

    glFlush();
}

int main(int argc, char** argv) {
    glutInit(&argc, argv);
    glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
    glutInitWindowSize(800, 900);
    glutCreateWindow("House-ComputerGraphicsLab-01");

    init();
    glutDisplayFunc(display);
    glutMainLoop();
    return 0;
}
```

## 5 Output

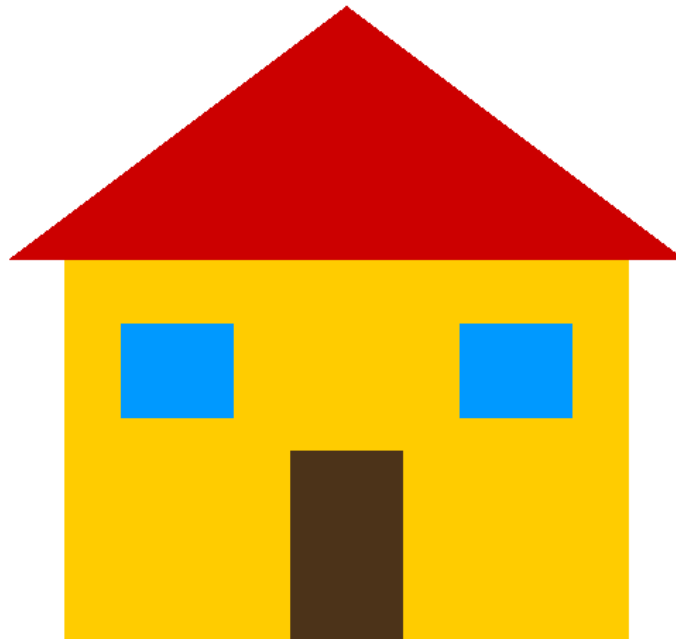


Figure 2: Output of House Drawing Using OpenGL

## 6 Discussion

This experiment demonstrates how simple geometric shapes can be combined to form complex objects using OpenGL. The use of orthographic projection simplifies the coordinate calculations and is suitable for 2D graphics applications.

## 7 Conclusion

In this lab, a house was successfully drawn using OpenGL and GLUT. The experiment helped in understanding basic OpenGL functions, coordinate systems, and drawing primitives. This knowledge forms the foundation for more advanced graphics programming.