**Lab Taks-3**

Submission Guidelines-

* Rename the file to your id only. If your id is 18-XXXXX-1, then the file name must be 18-XXXXX-1.docx.
* Must submit within time that will be discussed in class VUES to the section named Lab Tak-3
* Must include resources for all the section in the table

|  |
| --- |
| **Question- 1**  Draw five storied building with windows and a front door |
| **Graph Plot (Picture)-** |
| **Code-**  **#include <windows.h> // for MS Windows**  **#include <GL/glut.h> // GLUT, include glu.h and gl.h**  **void initGL() {**  **// Set "clearing" or background color**  **glClearColor(1.0f, 1.0f, 1.0f, 0.0f); // Black and opaque**  **}**  **void display() {**  **glClear(GL\_COLOR\_BUFFER\_BIT); // Clear the color buffer with current clearing color**  **glBegin(GL\_QUADS); // These vertices form a closed polygon**  **glColor3f(0.0f, 1.0f, 1.0f);**  **glVertex2f(0.3f, 0.9f);**  **glVertex2f(-0.3f, 0.9f);**  **glVertex2f(-0.3f, -0.8f);**  **glVertex2f(0.3f, -0.8f);**  **glEnd();**  **glFlush(); // Render now**  **glBegin(GL\_QUADS); // These vertices form a closed polygon**  **glColor3f(0.5f, 1.0f, 1.0f);**  **glVertex2f(0.1f, -0.5f);**  **glVertex2f(-0.1f, -0.5f);**  **glVertex2f(-0.1f, -0.8f);**  **glVertex2f(0.1f, -0.8f);**  **glEnd();**  **glFlush();**  **glBegin(GL\_QUADS); // These vertices form a closed polygon**  **glColor3f(1.0f, 1.0f, 0.0f); // Yellow**  **glVertex2f(0.1f, 0.8f);**  **glVertex2f(0.2f, 0.8f);**  **glVertex2f(0.2f, 0.7f);**  **glVertex2f(0.1f, 0.7f);**  **glEnd();**  **glFlush();**  **glBegin(GL\_QUADS); // These vertices form a closed polygon**  **glColor3f(1.0f, 1.0f, 0.0f); // Yellow**  **glVertex2f(-0.1f, 0.8f);**  **glVertex2f(-0.2f, 0.8f);**  **glVertex2f(-0.2f, 0.7f);**  **glVertex2f(-0.1f, 0.7f);**  **glEnd();**  **glFlush();**  **glBegin(GL\_QUADS); // These vertices form a closed polygon**  **glColor3f(1.0f, 1.0f, 0.0f); // Yellow**  **glVertex2f(0.1f, 0.6f);**  **glVertex2f(0.2f, 0.6f);**  **glVertex2f(0.2f, 0.5f);**  **glVertex2f(0.1f, 0.5f);**  **glEnd();**  **glFlush();**  **glBegin(GL\_QUADS); // These vertices form a closed polygon**  **glColor3f(1.0f, 1.0f, 0.0f); // Yellow**  **glVertex2f(-0.1f, 0.6f);**  **glVertex2f(-0.2f, 0.6f);**  **glVertex2f(-0.2f, 0.5f);**  **glVertex2f(-0.1f, 0.5f);**  **glEnd();**  **glFlush();**  **glBegin(GL\_QUADS); // These vertices form a closed polygon**  **glColor3f(1.0f, 1.0f, 0.0f); // Yellow**  **glVertex2f(0.1f, 0.4f);**  **glVertex2f(0.2f, 0.4f);**  **glVertex2f(0.2f, 0.3f);**  **glVertex2f(0.1f, 0.3f);**  **glEnd();**  **glFlush();**  **glBegin(GL\_QUADS); // These vertices form a closed polygon**  **glColor3f(1.0f, 1.0f, 0.0f); // Yellow**  **glVertex2f(-0.1f, 0.4f);**  **glVertex2f(-0.2f, 0.4f);**  **glVertex2f(-0.2f, 0.3f);**  **glVertex2f(-0.1f, 0.3f);**  **glEnd();**  **glFlush();**  **glBegin(GL\_QUADS); // These vertices form a closed polygon**  **glColor3f(1.0f, 1.0f, 0.0f); // Yellow**  **glVertex2f(0.1f, 0.2f);**  **glVertex2f(0.2f, 0.2f);**  **glVertex2f(0.2f, 0.1f);**  **glVertex2f(0.1f, 0.1f);**  **glEnd();**  **glFlush();**  **glBegin(GL\_QUADS); // These vertices form a closed polygon**  **glColor3f(1.0f, 1.0f, 0.0f); // Yellow**  **glVertex2f(-0.1f, 0.2f);**  **glVertex2f(-0.2f, 0.2f);**  **glVertex2f(-0.2f, 0.1f);**  **glVertex2f(-0.1f, 0.1f);**  **glEnd();**  **glFlush();**  **glBegin(GL\_QUADS); // These vertices form a closed polygon**  **glColor3f(1.0f, 1.0f, 0.0f); // Yellow**  **glVertex2f(0.1f, 0.0f);**  **glVertex2f(0.2f, 0.0f);**  **glVertex2f(0.2f, -0.1f);**  **glVertex2f(0.1f, -0.1f);**  **glEnd();**  **glFlush();**  **glBegin(GL\_QUADS); // These vertices form a closed polygon**  **glColor3f(1.0f, 1.0f, 0.0f); // Yellow**  **glVertex2f(-0.1f, 0.0f);**  **glVertex2f(-0.2f, 0.0f);**  **glVertex2f(-0.2f, -0.1f);**  **glVertex2f(-0.1f, -0.1f);**  **glEnd();**  **glFlush();**  **glBegin(GL\_QUADS); // These vertices form a closed polygon**  **glColor3f(1.0f, 1.0f, 0.0f); // Yellow**  **glVertex2f(0.1f, -0.2f);**  **glVertex2f(0.2f, -0.2f);**  **glVertex2f(0.2f, -0.3f);**  **glVertex2f(0.1f, -0.3f);**  **glEnd();**  **glFlush();**  **glBegin(GL\_QUADS); // These vertices form a closed polygon**  **glColor3f(1.0f, 1.0f, 0.0f); // Yellow**  **glVertex2f(-0.1f, -0.2f);**  **glVertex2f(-0.2f, -0.2f);**  **glVertex2f(-0.2f, -0.3f);**  **glVertex2f(-0.1f, -0.3f);**  **glEnd();**  **glFlush();**  **}**  **/\* Main function: GLUT runs as a console application starting at main() \*/**  **int main(int argc, char\*\* argv) {**  **glutInit(&argc, argv); // Initialize GLUT**  **glutCreateWindow(" five storied building with windows and a front door"); // Create window with the given title**  **glutInitWindowSize(420, 420); // Set the window's initial width & height**  **glutInitWindowPosition(50, 50); // Position the window's initial top-left corner**  **glutDisplayFunc(display); // Register callback handler for window re-paint event**  **initGL(); // Our own OpenGL initialization**  **glutMainLoop(); // Enter the event-processing loop**  **return 0;**  **}** |
| **Output Screenshot (Full Screen)-** |

|  |
| --- |
| **Question- 2**  Draw a tree |
| **Graph Plot (Picture)-** |
| **Code-**  **#include<windows.h>**  **#include <GL/glut.h>**  **void init(void)**  **{**  **glClearColor(1.0, 1.0, 1.0, 0.0); // Set display window colour to white**  **glMatrixMode(GL\_PROJECTION); // Set projection parameters**  **gluOrtho2D(0.0, 35.0, 45.0, 0.0);**  **}**  **void drawShapes(void)**  **{**  **glClear(GL\_COLOR\_BUFFER\_BIT); // Clear display window**  **//Set colour to black**  **glColor3f(0.0, 0.0, 0.0);**  **//Adjust the point size**  **glPointSize(5.0);**  **//Set colour to gray**  **glColor3f(0.5f, 0.5f, 0.0f);**  **// Draw an outlined triangle**  **glBegin(GL\_QUADS);**  **glVertex2i(16,35); //base**  **glVertex2i(19,35);**  **glVertex2i(19,25);**  **glVertex2i(16,25);**  **glEnd();**  **glColor3f(0.0f, 1.0f, 0.0f);**  **glBegin(GL\_POLYGON);**  **glVertex2i(10,25);**  **glVertex2i(25,25);**  **glVertex2i(17,18);**  **glEnd();**  **glColor3f(0.0f, 1.0f, 0.0f);**  **glBegin(GL\_POLYGON);**  **glVertex2i(10,22);**  **glVertex2i(25,22);**  **glVertex2i(17,15);**  **glEnd();**  **glColor3f(0.0f, 1.0f, 0.0f);**  **glBegin(GL\_POLYGON);**  **glVertex2i(11,19);**  **glVertex2i(25,19);**  **glVertex2i(17,12);**  **glEnd();**  **glFlush(); // Process all OpenGL routines**  **}**  **int main(int argc, char\* argv[])**  **{**  **glutInit(&argc, argv); // Initalise GLUT**  **glutInitDisplayMode(GLUT\_SINGLE|GLUT\_RGB); // Set display mode**  **glutInitWindowPosition(50, 100); // Set window position**  **glutInitWindowSize(400, 300); // Set window size**  **glutCreateWindow("An Example OpenGL Program"); // Create display window**  **init(); // Execute initialisation procedure**  **glutDisplayFunc(drawShapes); // Send graphics to display window**  **glutMainLoop(); // Display everything and wait**  **return 0;**  **}** |
| **Output Screenshot (Full Screen)-** |

|  |
| --- |
| **Question- 3**  Draw a lamppost with black background |
| **Graph Plot (Picture)-** |
| **Code-**  **#include <windows.h> // for MS Windows**  **#include <GL/glut.h> // GLUT, include glu.h and gl.h**  **/\* Initialize OpenGL Graphics \*/**  **void initGL() {**  **// Set "clearing" or background color**  **glClearColor(0.0f, 0.0f, 0.0f, 1.0f); // Black and opaque**  **}**  **/\* Handler for window-repaint event. Call back when the window first appears and**  **whenever the window needs to be re-painted. \*/**  **void display() {**  **glClear(GL\_COLOR\_BUFFER\_BIT); // Clear the color buffer with current clearing color**  **glBegin(GL\_QUADS); // Each set of 4 vertices form a quad**  **glColor3f(0.0f, 1.0f, 0.0f); // green**  **glVertex2f(-0.8f, 0.0f); // x, y**  **glVertex2f(-0.7f, 0.0f);**  **glVertex2f(-0.7f, 0.70f); // x, y**  **glVertex2f(-0.8f, 0.70f);**  **glEnd();**  **glBegin(GL\_QUADS); // Each set of 4 vertices form a quad**  **glColor3f(0.0f, 1.0f, 0.0f); // green**  **glVertex2f(-0.7f, 0.69f); // x, y**  **glVertex2f(-0.4f, 0.69f);**  **glVertex2f(-0.4f, 0.70f); // x, y**  **glVertex2f(-0.7f, 0.70f);**  **glEnd();**  **glBegin(GL\_QUADS); // Each set of 4 vertices form a quad**  **glColor3f(1.0f, 1.0f, 0.0f); // yellow**  **glVertex2f(-0.5f, 0.67f); // x, y**  **glVertex2f(-0.4f, 0.67f);**  **glVertex2f(-0.4f, 0.69f); // x, y**  **glVertex2f(-0.5f, 0.69f);**  **glEnd();**  **glFlush(); // Render now**  **}**  **/\* Main function: GLUT runs as a console application starting at main() \*/**  **int main(int argc, char\*\* argv) {**  **glutInit(&argc, argv); // Initialize GLUT**  **glutCreateWindow(" a lamp post"); // Create window with the given title**  **glutInitWindowSize(320, 320); // Set the window's initial width & height**  **glutDisplayFunc(display); // Register callback handler for window re-paint event**  **initGL(); // Our own OpenGL initialization**  **glutMainLoop(); // Enter the event-processing loop**  **return 0;**  **}** |
| **Output Screenshot (Full Screen)-** |

|  |
| --- |
| **Question- 4**  Draw a bench |
| **Graph Plot (Picture)-** |
| **Code-**  **#include <windows.h>**  **#include <GL/glut.h>**  **void display() {**  **glClearColor(1.0f, 1.0f, 1.0f, 1.0f);**  **glClear(GL\_COLOR\_BUFFER\_BIT);**  **glLineWidth(7.7);**  **glBegin(GL\_LINES);**  **glColor3f(0.0f, 0.0f, 0.0f);**  **glVertex2f(-0.0f, 0.1f);**  **glVertex2f(-0.0f,-0.15f);**  **glEnd();**  **glLineWidth(7.7);**  **glBegin(GL\_LINES);**  **glColor3f(0.0f, 0.0f, 0.0f);**  **glVertex2f(0.1f, 0.05f);**  **glVertex2f(0.1f,-0.2f);**  **glEnd();**  **glLineWidth(7.7);**  **glBegin(GL\_LINES);**  **glColor3f(0.0f, 0.0f, 0.0f);**  **glVertex2f(0.6f, 0.05f);**  **glVertex2f(0.6f,-0.2f);**  **glEnd();**  **glLineWidth(7.7);**  **glBegin(GL\_LINES);**  **glColor3f(0.0f, 0.0f, 0.0f);**  **glVertex2f(0.5f, 0.1f);**  **glVertex2f(0.5f,-0.15f);**  **glEnd();**  **glBegin(GL\_QUADS);**  **glColor3f(0.1f, 0.0f, 0.0f);**  **glVertex2f(-0.05f, 0.1f);**  **glVertex2f(0.1f, 0.0f);**  **glVertex2f(0.68f, 0.0f);**  **glVertex2f(0.57f, 0.1f);**  **glEnd();**  **glBegin(GL\_QUADS);**  **glColor3f(0.0f, 0.5f, 1.0f);**  **glVertex2f(-0.05f, 0.1f);**  **glVertex2f(0.57f, 0.1f);**  **glVertex2f(0.57f, 0.3f);**  **glVertex2f(-0.05f, 0.3f);**  **glEnd();**  **glFlush();**  **}**  **int main(int argc, char\*\* argv) {**  **glutInit(&argc, argv);**  **glutCreateWindow("a bench");**  **glutInitWindowSize(320, 320);**  **glutDisplayFunc(display);**  **glutMainLoop();**  **return 0;**  **}** |
| **Output Screenshot (Full Screen)-** |