**Lab Taks-1**

|  |
| --- |
| **Question-**  Draw the object- |
| **Graph Plot (Picture)-** |
| **Code-**  #include <windows.h> // for MS Windows  #include <GL/glut.h> // GLUT, include glu.h and gl.h  /\* Handler for window-repaint event. Call back when the window first appears and  whenever the window needs to be re-painted. \*/  void display() {  glClearColor(1.0f, 1.0f, 1.0f, 1.0f);  glClear(GL\_COLOR\_BUFFER\_BIT);  glLineWidth(4.5);  glBegin(GL\_LINES);  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(-0.7, -0.5);  glVertex2f(0.7, -0.5);  glVertex2f(0.7, -0.5);  glVertex2f(0.7, 0.5);  glVertex2f(0.7, 0.5);  glVertex2f(-0.7, 0.5);  glVertex2f(-0.7, 0.5);  glVertex2f(-0.7, -0.5);  glEnd();  glFlush(); // Render now  }  /\* Main function: GLUT runs as a console application starting at main() \*/  int main(int argc, char\*\* argv) {  glutInitWindowSize(720, 620); // Set the window's initial width & height  glutInit(&argc, argv); // Initialize GLUT  glutCreateWindow("OpenGL Setup"); // Create a window with the given title  glutDisplayFunc(display); // Register display callback handler for window re-paint  glutMainLoop(); // Enter the event-processing loop  return 0;  } |
| **Output Screenshot (Full Screen)-** |

|  |
| --- |
| **Question-**  Draw the object- |
| **Graph Plot (Picture)-** |
| **Code-**  //Draw Line  #include <windows.h> // for MS Windows  #include <GL/glut.h> // GLUT, include glu.h and gl.h  /\* Handler for window-repaint event. Call back when the window first appears and  whenever the window needs to be re-painted. \*/  void display() {  glClearColor(1.0f, 1.0f, 1.0f, 1.0f); // Set background color to black and opaque  glClear(GL\_COLOR\_BUFFER\_BIT); // Clear the color buffer (background)  glLineWidth(3.5);  glBegin(GL\_POLYGON);  glColor3f(1.0f, 0.0f, 0.0f); // Red  glVertex2f(-0.7, -0.5);  glVertex2f(0.7, -0.5);  glVertex2f(0.4, 0.5);  glVertex2f(-0.4, 0.5);  glEnd();  glFlush();  }  /\* Main function: GLUT runs as a console application starting at main() \*/  int main(int argc, char\*\* argv) {  glutInitWindowSize(620, 520); // Set the window's initial width & height  glutInit(&argc, argv); // Initialize GLUT  glutCreateWindow("OpenGL Setup"); // Create a window with the given title  glutDisplayFunc(display); // Register display callback handler for window re-paint  glutMainLoop(); // Enter the event-processing loop  return 0;  } |
| **Output Screenshot (Full Screen)-**  **A screenshot of a computer  Description automatically generated** |

|  |
| --- |
| **Question-**  Draw the object-  Octagon Shape | Area & Angles - Video & Lesson Transcript | Study.com |
| **Graph Plot (Picture)-** |
| **Code-**  //Draw Line  #include <windows.h> // for MS Windows  #include <GL/glut.h> // GLUT, include glu.h and gl.h  /\* Handler for window-repaint event. Call back when the window first appears and  whenever the window needs to be re-painted. \*/  void polygon(){  }  void display() {  glClearColor(1.0f, 1.0f, 1.0f, 1.0f); // Set background color to black and opaque  glClear(GL\_COLOR\_BUFFER\_BIT); // Clear the color buffer (background)  glLineWidth(3.5);  glBegin(GL\_POLYGON);  glColor3f(1.0f, 0.0f, 0.0f); // Red  glVertex2f(-0.5, -0.3); // x, y  glVertex2f(-0.2, -0.6);  glVertex2f(0.2, -0.6);  glVertex2f(0.5, -0.3);  glVertex2f(0.5, 0.3);  glVertex2f(0.2, 0.6);  glVertex2f(-0.2, 0.6);  glVertex2f(-0.5, 0.3);  glEnd();  glBegin(GL\_LINES);  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(-0.5, -0.3);  glVertex2f(-0.2, -0.6);  glVertex2f(-0.2, -0.6);  glVertex2f(0.2, -0.6);  glVertex2f(0.2, -0.6);  glVertex2f(0.5, -0.3);  glVertex2f(0.5, -0.3);  glVertex2f(0.5, 0.3);  glVertex2f(0.5, 0.3);  glVertex2f(0.2, 0.6);  glVertex2f(0.2, 0.6);  glVertex2f(-0.2, 0.6);  glVertex2f(-0.2, 0.6);  glVertex2f(-0.5, 0.3);  glVertex2f(-0.5, 0.3);  glVertex2f(-0.5, -0.3);  glEnd();  glFlush(); // Render now  }  /\* Main function: GLUT runs as a console application starting at main() \*/  int main(int argc, char\*\* argv) {  glutInitWindowSize(620, 520); // Set the window's initial width & height  glutInit(&argc, argv); // Initialize GLUT  glutCreateWindow("OpenGL Setup"); // Create a window with the given title  glutDisplayFunc(display); // Register display callback handler for window re-paint  glutMainLoop(); // Enter the event-processing loop  return 0;  } |
| **Output Screenshot (Full Screen)-** |

|  |
| --- |
| **Question-**  Draw the object- |
| **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 xy\_axis(){  glLineWidth(4);  glBegin(GL\_LINES);  glColor3f(0.0f, 0.0f, 0.0f);  glVertex2f(0.0f, 0.0f);  glVertex2f(1.0f, 0.0f);  glVertex2f(0.0f, 0.0f);  glVertex2f(0.0f, 1.0f);  glVertex2f(0.0f, 0.0f);  glVertex2f(-1.0f, 0.0f);  glVertex2f(0.0f, 0.0f);  glVertex2f(0.0f, -1.0f);  glEnd();  }  void Arrow(){  glBegin(GL\_POLYGON);  glColor3ub(67,156,73);  glVertex2f(0.1f, 0.3f);  glVertex2f(0.5f, 0.3f);  glVertex2f(0.5f, 0.5f);  glVertex2f(0.1f, 0.5f);  glEnd();  glBegin(GL\_POLYGON);  glColor3ub(67,156,73);  glVertex2f(0.5f, 0.2f);  glVertex2f(0.7f, 0.4f);  glVertex2f(0.5f, 0.6f);  glEnd();  }  void yellow\_triangle()  {  glBegin(GL\_TRIANGLES);  glColor3ub(255, 255, 20);  glVertex2f(+0.4f, -0.1f);  glVertex2f(+0.1f,-0.6f);  glVertex2f(+0.7f, -0.6f);  glEnd();  }  void purple\_triangle()  {  glBegin(GL\_TRIANGLES);  glColor3ub(112,36,125);  glVertex2f(-0.3f, -0.1f); // x, y  glVertex2f(-0.3f, -0.7f);  glVertex2f(-0.7f, -0.4f); // x, y  glEnd();  }  void red\_rectangle()  {  glBegin(GL\_POLYGON);  glColor3f(1.0f, 0.0f, 0.0f);  glVertex2f(-0.7, 0.3);  glVertex2f(-0.3, 0.3);  glVertex2f(-0.3, 0.6);  glVertex2f(-0.7, 0.6);  glEnd();  }  void initGL() {  // Set "clearing" or background color  glClearColor(0.0f, 0.0f, 0.0f, 1.0f); // Black and opaque  }  void display() {  glClearColor(1.0f, 1.0f, 1.0f, 1.0f);  glClear(GL\_COLOR\_BUFFER\_BIT); // Clear the color buffer with current clearing color  xy\_axis();  Arrow();  yellow\_triangle();  purple\_triangle();  red\_rectangle();  glFlush();  }  /\* Main function: GLUT runs as a console application starting at main() \*/  int main(int argc, char\*\* argv) {  glutInit(&argc, argv); // Initialize GLUT  glutInitWindowSize(620, 520); // Set the window's initial width & height  glutCreateWindow("Task 4"); // Create window with the given title  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)-**  **A screenshot of a computer  Description automatically generated** |