**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); // Set background color to white and opaque  glClear(GL\_COLOR\_BUFFER\_BIT); // Clear the color buffer (background)  glLineWidth(.5);  glBegin(GL\_LINES);  glColor3f(0.0f, 0.0f, 0.0f);//black  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();  glBegin(GL\_QUADS);  glColor3f(0.0f, 1.0f, 0.0f);//green  glVertex2f(-0.6f, 0.6f);  glVertex2f(-0.6f, 0.2f);  glVertex2f(-0.2f, 0.2f);  glVertex2f(-0.2f, 0.6f);  glEnd();  glBegin(GL\_TRIANGLES);  glColor3f(0.0f, 0.0f, 1.0f);//blue  glVertex2f(-0.2f, -0.1f);  glVertex2f(-0.6f, -0.3f);  glVertex2f(-0.2f,-0.5f);  glEnd();  glBegin(GL\_TRIANGLES);  glColor3f(1.0f, 1.0f, 0.0f);//yellow  glVertex2f(0.45f, -0.2f);  glVertex2f(0.2f, -0.4f);  glVertex2f(0.7f, -0.4f);  glEnd();  //arrow  /\*glBegin(GL\_TRIANGLES);//  glColor3ub(10, 100, 20);//rgb  glVertex2f(0.4,0.5f);  glVertex2f(0.5,0.35f);  glVertex2f(0.4f,0.2f);  glEnd();  glBegin(GL\_QUADS);  glColor3ub(10, 100, 20);//rgb  glVertex2f(0.2f,0.4f);  glVertex2f(0.4f,0.4f);  glVertex2f(0.4f,0.3f);  glVertex2f(0.2f,0.3f);  glEnd();\*/  //arrow  glBegin(GL\_POLYGON);  glColor3f(0.8f, 1.0f, 0.8f);//light green  glVertex2f(0.4f, 0.4f);  glVertex2f(0.2f, 0.4f);  glVertex2f(0.2f, 0.3f);  glVertex2f(0.4f, 0.3f);  glVertex2f(0.4f, 0.2f);  glVertex2f(0.5f, 0.35f);  glVertex2f(0.4f, 0.5f);  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("OpenGL Setup"); // Create a window with the given title  glutInitWindowSize(420, 420); // Set the window's initial width & height  glutDisplayFunc(display); // Register display callback handler for window re-paint  glutMainLoop(); // Enter the event-processing loop  return 0;  } |
| **Output Screenshot (Full Screen)-** |