// Name – Dahiwal Satyam Santoshkumar

// Roll Number – 207019

#include <GL/glut.h>

#include <stdlib.h>

#include <stdio.h>

#include <math.h>

#define RADDEG 57.29577951f

float XUP[3] = {1,0,0},

YUP[3] = {0,1,0},

ZUP[3] = {0,0,1},

ORG[3] = {0,0,0};

GLfloat viewangle = 0, tippangle = 0;

GLfloat d[3] = {0.1, 0.1, 0.1};

GLfloat xAngle = 0.0, yAngle = 0.0, zAngle = 0.0;

// Use arrow keys to rotate entire scene

void Special\_Keys (int key, int x, int y)

{

switch (key)

{

case GLUT\_KEY\_LEFT : viewangle -= 5; break;

case GLUT\_KEY\_RIGHT: viewangle += 5; break;

case GLUT\_KEY\_UP : tippangle -= 5; break;

case GLUT\_KEY\_DOWN : tippangle += 5; break;

default: printf (" Special key %c == %d\n", key, key);

}

glutPostRedisplay();

}

void Keyboard (unsigned char key, int x, int y)

{

switch (key)

{

case 'b' : d[0] += 0.1; break; //x

case 'n' : d[1] += 0.1; break; //y

case 'm' : d[2] += 0.1; break; //z

case 'x' : xAngle += 5; break;

case 'y' : yAngle += 5; break;

case 'z' : zAngle += 5; break;

default: printf (" Keyboard %c == %d\n", key, key);

}

glutPostRedisplay();

}

void Triad (void)

{

glColor3f (1.0, 1.0, 1.0);

glBegin (GL\_LINES);

glVertex3fv (ORG);

glVertex3fv (XUP); //x-axix

glVertex3fv (ORG);

glVertex3fv (YUP); //y-axix

glVertex3fv (ORG);

glVertex3fv (ZUP); //z-axix

glEnd ();

glRasterPos3f (1.1, 0.0, 0.0);

glutBitmapCharacter (GLUT\_BITMAP\_HELVETICA\_18, 'X');

glRasterPos3f (0.0, 1.1, 0.0);

glutBitmapCharacter (GLUT\_BITMAP\_HELVETICA\_18, 'Y');

glRasterPos3f (0.0, 0.0, 1.1);

glutBitmapCharacter (GLUT\_BITMAP\_HELVETICA\_18, 'Z');

}

void Draw\_Box (void)

{

glBegin (GL\_QUADS);

glColor3f ( 0.0, 0.7, 0.1); // Front - green

glVertex3f (-1.0, 1.0, 1.0);

glVertex3f ( 1.0, 1.0, 1.0);

glVertex3f ( 1.0, -1.0, 1.0);

glVertex3f (-1.0, -1.0, 1.0);

glColor3f ( 0.9, 1.0, 0.0); // Back - yellow

glVertex3f (-1.0, 1.0, -1.0);

glVertex3f ( 1.0, 1.0, -1.0);

glVertex3f ( 1.0, -1.0, -1.0);

glVertex3f (-1.0, -1.0, -1.0);

glColor3f ( 0.2, 0.2, 1.0); // Top - blue

glVertex3f (-1.0, 1.0, 1.0);

glVertex3f ( 1.0, 1.0, 1.0);

glVertex3f ( 1.0, 1.0, -1.0);

glVertex3f (-1.0, 1.0, -1.0);

glColor3f ( 0.7, 0.0, 0.1); // Bottom - red

glVertex3f (-1.0, -1.0, 1.0);

glVertex3f ( 1.0, -1.0, 1.0);

glVertex3f ( 1.0, -1.0, -1.0);

glVertex3f (-1.0, -1.0, -1.0);

glEnd();

}

void redraw (void)

{

int v;

glClear (GL\_COLOR\_BUFFER\_BIT | GL\_DEPTH\_BUFFER\_BIT);

glEnable (GL\_DEPTH\_TEST);

glLoadIdentity ();

glTranslatef (0, 0, -3);

glRotatef (tippangle, 1,0,0); // Up and down arrow keys 'tip' view.

glRotatef (viewangle, 0,1,0); // Right/left arrow keys 'turn' view.

// glDisable (GL\_LIGHTING);

Triad ();

// glPushMatrix ();

glTranslatef (d[0], d[1], d[2]);

// Move box down X axis.

glScalef (0.2, 0.2, 0.2);

glRotatef (zAngle, 0,0,1);

glRotatef (yAngle, 0,1,0);

glRotatef (xAngle, 1,0,0);

Draw\_Box ();

//glPopMatrix ();

glutSwapBuffers();

}

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

{

glutInit(&argc, argv);

glutInitWindowSize (900, 600);

glutInitWindowPosition (300, 300);

glutInitDisplayMode (GLUT\_DEPTH | GLUT\_DOUBLE);

glutCreateWindow ("3D Cube Rotation(press arrows and x y z keys)");

glutDisplayFunc( redraw );

glutKeyboardFunc ( Keyboard );

glutSpecialFunc (Special\_Keys);

glMatrixMode (GL\_PROJECTION)

gluPerspective (60, 1.5, 1, 10)

glMatrixMode (GL\_MODELVIEW)

glutMainLoop ();

return 1;

}

Output

satyam@ubuntu:~$ g++ eight.cpp -lglut -lGL -lGLEW -lGLU -o eight

satyam@ubuntu:~$ ./eight

