```
#include<stdlib.h>
#include<GL/glut.h>
GLfloat vertices[]= {-1, -1, -1,
                                    1, -1, -1,
                                    1, 1, -1,
                                    -1, 1, -1,
                                    -1, -1, 1,
                                    1, -1, 1,
                                    1, 1, 1,
                                    -1, 1, 1
                                };

GLfloat colors[]= {0, 0, 0,
                                1, 0, 0,
                                1, 1, 0,
                                0, 1, 0,
                                0, 0, 1,
                                1, 0, 1,
                                1, 1, 1,
                                0, 1, 1
                            };

GLubyte cubeIndices[]= {0, 3, 2, 1,
                                                2, 3, 7, 6,
                                                0, 4, 7, 3,
                                                1, 2, 6, 5,
                                                4, 5, 6, 7,
                                                0, 1, 5, 4
                                                };

static GLfloat theta[]= {0,0,0};
static GLint axis=2;

void display(void)
{
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);
    glLoadIdentity();

    glRotatef(theta[0], 1, 0, 0);
    glRotatef(theta[1], 0, 1, 0);
    glRotatef(theta[2], 0, 0, 1);

    glDrawElements(GL_QUADS,24,GL_UNSIGNED_BYTE,cubeIndices);

    glutSwapBuffers();
}

void spinCube()
{
    theta[axis] += 2;

    if(theta[axis] > 360)
        theta[axis] -= 360;

    glutPostRedisplay();
}

void mouse(int btn, int state, int x, int y)
{
    if(btn==GLUT_LEFT_BUTTON && state==GLUT_DOWN)
        axis=0;
    if(btn==GLUT_MIDDLE_BUTTON && state==GLUT_DOWN)
        axis=1;
    if(btn==GLUT_RIGHT_BUTTON && state==GLUT_DOWN)
        axis=2;
}

void myReshape(int w, int h)
{
    glViewport(0,0,w,h);
    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
```

```
    if(w<=h)
        glOrtho(-2, 2, -2*(GLfloat)h/(GLfloat)w, 2*(GLfloat)h/(GLfloat)w, -10, 10);
    else
        glOrtho(-2*(GLfloat)w/(GLfloat)h, 2*(GLfloat)w/(GLfloat)h,  -2, 2, -10, 10);
    glMatrixMode(GL_MODELVIEW);
}

int main(int argc, char **argv)
{
    glutInit(&argc, argv);
    glutInitDisplayMode(GLUT_DOUBLE|GLUT_RGB|GLUT_DEPTH);
    glutInitWindowSize(500, 500);
    glutCreateWindow("Spin a color cube");

    glutReshapeFunc(myReshape);
    glutDisplayFunc(display);
    glutIdleFunc(spinCube);
    glutMouseFunc(mouse);

    glEnable(GL_DEPTH_TEST);

    glEnableClientState(GL_COLOR_ARRAY);
    glEnableClientState(GL_VERTEX_ARRAY);

    glVertexPointer(3, GL_FLOAT, 0, vertices);
    glColorPointer(3, GL_FLOAT, 0, colors);

    glColor3f(1, 1, 1);

    glutMainLoop();
}
```