

```
#include<windows.h>
#include <GL/glut.h>
```

```
float angle = 0.0f;
float translateX = 0.0f;
float translateY = 0.0f;
```

```
void drawSquare()
{
    glClear(GL_COLOR_BUFFER_BIT);
    glMatrixMode(GL_MODELVIEW);
    glLoadIdentity();
    glTranslatef(translateX, translateY, -1.0f);
    glRotatef(angle, 1.0f, 0.0f, 0.0f);
    glColor3f(1.0f, 0.0f, 0.0f);
    glBegin(GL_QUADS);
        glVertex2f(-0.5f, -0.5f);
        glVertex2f(0.5f, -0.5f);
        glVertex2f(0.5f, 0.5f);
        glVertex2f(-0.5f, 0.5f);
    glEnd();
    glutSwapBuffers();
}
```

```
void handleKeypress(unsigned char key, int x, int y)
{
    switch(key)
    {
        case 27: // escape key
            exit(0);
            break;
        case 'a':
            translateX -= 0.1f;
            break;
        case 'd':
            translateX += 0.1f;
            break;
        case 'w':
            translateY += 0.1f;
            break;
        case 's':
            translateY -= 0.1f;
            break;
    }
}
```

```

    glutPostRedisplay();
}

void handleSpecialKeypress(int key, int x, int y)
{
    switch(key)
    {
        case GLUT_KEY_LEFT:
            translateX -= 0.1f;
            break;
        case GLUT_KEY_RIGHT:
            translateX += 0.1f;
            break;
        case GLUT_KEY_UP:
            translateY += 0.1f;
            break;
        case GLUT_KEY_DOWN:
            translateY -= 0.1f;
            break;
    }
    glutPostRedisplay();
}

void handleMouse(int button, int state, int x, int y)
{
    if (button == GLUT_LEFT_BUTTON && state == GLUT_DOWN)
    {
        angle += 10.0f;
        if (angle > 360.0f)
            angle -= 360.0f;
        glutPostRedisplay();
    }
}

void update(int value)
{
    angle += 5.0f;
    if (angle > 360.0f)
        angle -= 360.0f;
    glutPostRedisplay();
    glutTimerFunc(25, update, 0);
}

```

```
int main(int argc, char **argv)
{
    glutInit(&argc, argv);
    glutInitDisplayMode(GLUT_DOUBLE);
    glutInitWindowSize(500, 500);
    glutCreateWindow("Rotating Square");
    glutTimerFunc(25, update, 0);
    glutDisplayFunc(drawSquare);
    glutKeyboardFunc(handleKeypress);
    glutSpecialFunc(handleSpecialKeypress);
    glutMouseFunc(handleMouse);

    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
    glOrtho(-1.0f, 1.0f, -1.0f, 1.0f, -1.0f, 1.0f);

    glutMainLoop();
    return 0;
}
```