## Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського» Факультет інформатики та обчислювальної техніки Кафедра інформаційних систем та технологій

## Лабораторна робота № 6

з дисципліни «Технології та засоби розробки комп'ютерної графіки та мультимедіа»

Тема: « Побудава та переміщення графічних об'єктів в просторі з використанням ОрепGL. »

Виконав:	Перевірила:
студент групи IO-25	ст. вик. кафедри ІСТ
Льоскін Іван Вадимович	Хмелюк Марина Сергіївна
Дата здачі	
Захищено з балом	

## **Завдання:** Побудава та переміщення графічних об'єктів в просторі з використанням OpenGL

## Код програми:

```
using System;
using System.Drawing;
using SharpGL;
using System. Windows. Forms;
namespace WindowsFormsApp6
   public partial class Form1 : Form
       private float rquad = 0;
       private float speed = 2f;
        private float savedSpeed = 2f;
        private float xTranslation = 0;
        private float yTranslation = 0;
        private float zTranslation = -7f;
        private float xScale = 1;
        private float yScale = 1;
        private float zScale = 1;
        private Timer renderTimer;
        public Form1()
        {
            InitializeComponent();
            this.KeyDown += Form1 KeyDown;
            this.KeyPreview = true;
            renderTimer = new Timer();
            renderTimer.Interval = 16;
            renderTimer.Tick += (s, e) => { openGLControl1 Load(null, null);
};
            renderTimer.Start();
        }
        private void Form1 KeyDown(object sender, KeyEventArgs e)
            switch (e.KeyCode)
                case Keys.Q:
                    speed -= 0.1f;
                    if (speed < Of) speed = Of;
                    break;
                case Keys.E:
                    speed += 0.1f;
                    break;
                case Keys.F:
                    if (speed > 0)
                        savedSpeed = speed;
                        speed = 0f;
```

```
else
                    {
                        speed = savedSpeed;
                    break;
                case Keys.W:
                    yTranslation += 0.5f;
                    break;
                case Keys.S:
                    yTranslation -= 0.5f;
                    break;
                case Keys.A:
                    xTranslation -= 0.5f;
                    break;
                case Keys.D:
                    xTranslation += 0.5f;
                    break;
                case Keys.Z:
                    zTranslation -= 0.5f;
                    break;
                case Keys.X:
                    zTranslation += 0.5f;
                    break;
                case Keys.C:
                    xScale += 0.1f;
                    yScale += 0.1f;
                    zScale += 0.1f;
                    break;
                case Keys.V:
                    if (xScale > 0.5f)
                        xScale -= 0.1f;
                        yScale -= 0.1f;
                        zScale -= 0.1f;
                        break;
                    break;
           }
        }
        private void Form1 Load(object sender, EventArgs e)
            this.Focus();
        private void openGLControl1_Load(object sender,
SharpGL.RenderEventArgs args)
        {
            OpenGL gL = this.openGLControl1.OpenGL;
            gL.Clear(OpenGL.GL_COLOR_BUFFER_BIT |
OpenGL.GL_DEPTH_BUFFER_BIT);
```

}

```
gL.LoadIdentity();
gL.Translate(xTranslation, yTranslation, zTranslation);
gL.Scale(xScale, yScale, zScale);
gL.Rotate(rquad, 1.0f, -1.0f, 1.0f);
gL.Begin(OpenGL.GL QUADS);
// front face of the cube
gL.Color(1.0f, 0.0f, 0.0f);
gL.Vertex(-1.0f, -1.0f, 1.0f);
gL.Vertex(1.0f, -1.0f, 1.0f);
gL.Vertex(1.0f, 1.0f, 1.0f);
gL.Vertex(-1.0f, 1.0f, 1.0f);
// back face of the cube
gL.Color(1.0f, 1.0f, 0.0f);
gL.Vertex(-1.0f, -1.0f, -1.0f);
gL.Vertex(-1.0f, 1.0f, -1.0f);
gL. Vertex (1.0f, 1.0f, -1.0f);
gL.Vertex(1.0f, -1.0f, -1.0f);
// left face of the cube
gL.Color(1.0f, 1.0f, 1.0f);
gL.Vertex(-1.0f, -1.0f, -1.0f);
gL.Vertex(-1.0f, -1.0f, 1.0f);
gL.Vertex(-1.0f, 1.0f, 1.0f);
gL.Vertex(-1.0f, 1.0f, -1.0f);
// right face of the cube
gL.Color(0.0f, 1.0f, 1.0f);
gL.Vertex(1.0f, -1.0f, -1.0f);
gL.Vertex(1.0f, -1.0f, 1.0f);
gL.Vertex(1.0f, 1.0f, 1.0f);
gL. Vertex (1.0f, 1.0f, -1.0f);
// top face of the cube
gL.Color(0.0f, 0.0f, 1.0f);
gL.Vertex(-1.0f, 1.0f, -1.0f);
gL.Vertex(-1.0f, 1.0f, 1.0f);
gL.Vertex(1.0f, 1.0f, 1.0f);
gL. Vertex (1.0f, 1.0f, -1.0f);
// bottom face of the cube
gL.Color(0.0f, 1.0f, 0.0f);
gL.Vertex(-1.0f, -1.0f, -1.0f);
gL.Vertex(1.0f, -1.0f, -1.0f);
gL.Vertex(1.0f, -1.0f, 1.0f);
gL.Vertex(-1.0f, -1.0f, 1.0f);
qL.End();
```

```
gL.Flush();

rquad += speed;
DrawText($"Rotation Angle: {speed:F2} \nCube coordinates:
{xTranslation:F2} {yTranslation:F2} {zTranslation:F2}");
}

private void DrawText(string text)
{
    using (Graphics g = openGLControl1.CreateGraphics())
    {
        Font font = new Font("Arial", 12);
        Brush brush = Brushes.White;
        g.DrawString(text, font, brush, new PointF(10, 10));
    }
}
}
```

**Висновок:** Виконавши цю лабораторну роботу я навчився стереометрії у комп'ютерній графіці. Мені довелось маніпулювати розмірами, положенням та обертанням геометричної фігури куб у просторі. Найбільшою складністю з якою мені довелося зіткнутися була проблема у тому щоб збагнути теоретичні відомості та робота з новими пакетами.