# **Лабораторная работа №6**

Условие:

*Лабораторная работа 6a. Построение и визуализация трехмерных объектов.*

Выполнение настоящей работы имеет целью закрепление теоретического материала и практическое освоение основных методов и алгоритмов трехмерной визуализации. Задача: Написать приложение/веб-приложение, в котором формируется и визуализируется заданный трехмерный объект (каркасная модель). Варианты: В качестве трехмерного объекта берется трехмерное изображение первой буквы фамилии.

*Лабораторная работа 6b. Реализация трехмерных преобразований.*

Задача: Дополнить приложение/веб-приложение, реализовав трехмерные преобразования заданного объекта (масштабирование, перенос, вращение вокруг произвольной оси).

*Лабораторная работа 6c. Построение простейших проекций.*

Задача: Дополнить приложение/веб-приложение, реализовав построение трех ортографических проекций заданного объекта (на координатные плоскости Oxy, Oxz, Oyz).

Требования и критерии оценки:

• Для задания трехмерного объекта использовать массив с координатами вершин.

• Графический интерфейс (система координат, оси)

• При выполнении трехмерных преобразований выводить итоговую матрицу преобразования.

# Ход решения:

Создаем форму, на нее кидаем компонент GLControl

Инициализируем его

GL.ClearColor(0.9f, 0.9f, 0.9f, 1);

Создаем матрицу проекции

GL.MatrixMode(MatrixMode.Projection);

GL.LoadIdentity();

Matrix4 m;

m =

Matrix4.LookAt(0, 1, 3, 0, 0, 0, 0, 1, 0) \*

Matrix4.CreatePerspectiveFieldOfView((float)(60 \* Math.PI / 180.0), 1, 0.1f, 10);

GL.LoadMatrix(ref m);

Затем каждый кадр очищаем кадр

GL.Clear(ClearBufferMask.ColorBufferBit | ClearBufferMask.DepthBufferBit);

Создаем матрицу поворота текущего обьекта

GL.MatrixMode(MatrixMode.Modelview);

Matrix4 m =

Matrix4.CreateTranslation( -3.4f, -4.4f, 0 ) \*

Matrix4.CreateRotationX(rb) \*

Matrix4.CreateRotationY(ra) \*

Matrix4.CreateScale( 0.2f, 0.2f, 0.2f );

И рисуем

GL.Begin(PrimitiveType.Lines);

for (int i = 0; i < vert.Length - 1; i++)

{

GL.Vertex3(vert[i].X, vert[i].Y, 0);

GL.Vertex3(vert[i+1].X, vert[i+1].Y, 0);

GL.Vertex3(vert[i].X, vert[i].Y, 3);

GL.Vertex3(vert[i + 1].X, vert[i + 1].Y, 3);

GL.Vertex3(vert[i].X, vert[i].Y, 0);

GL.Vertex3(vert[i].X, vert[i].Y, 3);

}

GL.End();

Буква задана как один слой вершин. Объем создается алгоритмом.

Вывод программы:

