Министерство образования и науки Российской Федерации

**Федеральное государственное бюджетное образовательное учреждение науки и высшего образования**

«Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»

(ВлГУ)

Кафедра информационных систем и программной инженерии

**Лабораторная работа № 13**

**по дисциплине**

**«Программирование компьютерной графики»**

**Тема: «Текстуры»**

Выполнил:

ст. гр. ПРИ-117

Хлызова В.Г.

Принял:

Жигалов И.Е.

Владимир, 2021

ЦЕЛЬ РАБОТЫ

Изучение способов реализации стилей заполнения поверхностей и закрашивания, которые имитируют сложную рельефную объемную поверхность, выполненную из определенного материала.

ЗАДАНИЕ

1. Ознакомиться по методическим указаниям и литературе с теоретическим материалом.

2. Выполнить действия, приведенные в разделе 13.2. При разработке программы имя проекта, создаваемого в MS Visual Studio, должно содержать фамилию студента и группу (например, Ivanov\_Ivan\_ISG\_105\_lab\_1).

3. При выполнении 13.2 использовать загрузку текстуры с именем, фамилией и группой студента, выполнявшего работу.

ХОД РАБОТЫ

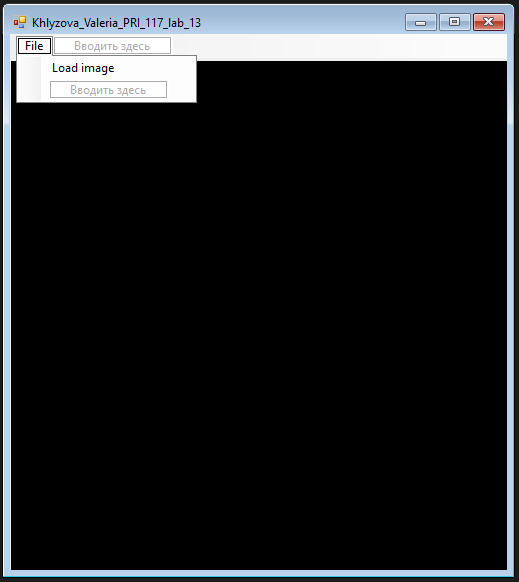


Рисунок 1 - Конструктор формы.

using System;

using System.IO;

using System.Windows.Forms;

using Tao.DevIl;

using Tao.FreeGlut;

using Tao.OpenGl;

namespace Khlyzova\_Valeria\_PRI\_117\_lab\_13

{

public partial class Form1 : Form

{

private int rot = 0;

private bool textureIsLoad = false;

public string texture\_name = "";

public int imageId = 0;

public uint mGlTextureObject = 0;

public int myImageId = 0;

public uint mGlTextureMyObject = 1;

public Form1()

{

InitializeComponent();

AnT.InitializeContexts();

}

private void Form1\_Load(object sender, EventArgs e)

{

Glut.glutInit();

Glut.glutInitDisplayMode(Glut.GLUT\_RGB | Glut.GLUT\_DOUBLE);

Il.ilInit();

Il.ilEnable(Il.IL\_ORIGIN\_SET);

Gl.glClearColor(255, 255, 255, 1);

Gl.glViewport(0, 0, AnT.Width, AnT.Height);

Gl.glMatrixMode(Gl.GL\_PROJECTION);

Gl.glLoadIdentity();

Glu.gluPerspective(30, AnT.Width / AnT.Height, 1, 100);

Gl.glMatrixMode(Gl.GL\_MODELVIEW);

Gl.glLoadIdentity();

Gl.glEnable(Gl.GL\_DEPTH\_TEST);

Gl.glEnable(Gl.GL\_LIGHTING);

Gl.glEnable(Gl.GL\_LIGHT0);

LoadPicture();

RenderTimer.Start();

}

private void RenderTimer\_Tick(object sender, EventArgs e)

{

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

Gl.glClearColor(255, 255, 255, 1);

Gl.glLoadIdentity();

Gl.glEnable(Gl.GL\_TEXTURE\_2D);

Gl.glPushMatrix();

Draw();

DrawIndividualTask();

AnT.Invalidate();

Gl.glPopMatrix();

Gl.glDisable(Gl.GL\_TEXTURE\_2D);

}

private void Draw()

{

if (textureIsLoad)

{

rot++;

if (rot > 360)

rot = 0;

Gl.glBindTexture(Gl.GL\_TEXTURE\_2D, mGlTextureObject);

Gl.glTranslated(0, -0.5, 0);

Gl.glRotated(rot, 0, 1, 0);

Gl.glBegin(Gl.GL\_QUADS);

Gl.glVertex3d(1, 1, 0);

Gl.glTexCoord2f(0, 0);

Gl.glVertex3d(1, 0, 0);

Gl.glTexCoord2f(1, 0);

Gl.glVertex3d(0, 0, 0);

Gl.glTexCoord2f(1, 1);

Gl.glVertex3d(0, 1, 0);

Gl.glTexCoord2f(0, 1);

Gl.glEnd();

}

}

private static uint MakeGlTexture(int Format, IntPtr pixels, int w, int h)

{

uint texObject;

Gl.glGenTextures(1, out texObject);

Gl.glPixelStorei(Gl.GL\_UNPACK\_ALIGNMENT, 1);

Gl.glBindTexture(Gl.GL\_TEXTURE\_2D, texObject);

Gl.glTexParameteri(Gl.GL\_TEXTURE\_2D, Gl.GL\_TEXTURE\_WRAP\_S, Gl.GL\_REPEAT);

Gl.glTexParameteri(Gl.GL\_TEXTURE\_2D, Gl.GL\_TEXTURE\_WRAP\_T, Gl.GL\_REPEAT);

Gl.glTexParameteri(Gl.GL\_TEXTURE\_2D, Gl.GL\_TEXTURE\_MAG\_FILTER, Gl.GL\_LINEAR);

Gl.glTexParameteri(Gl.GL\_TEXTURE\_2D, Gl.GL\_TEXTURE\_MIN\_FILTER, Gl.GL\_LINEAR);

Gl.glTexEnvf(Gl.GL\_TEXTURE\_ENV, Gl.GL\_TEXTURE\_ENV\_MODE, Gl.GL\_REPLACE);

switch (Format)

{

case Gl.GL\_RGB:

Gl.glTexImage2D(Gl.GL\_TEXTURE\_2D, 0, Gl.GL\_RGB, w, h, 0, Gl.GL\_RGB, Gl.GL\_UNSIGNED\_BYTE, pixels);

break;

case Gl.GL\_RGBA:

Gl.glTexImage2D(Gl.GL\_TEXTURE\_2D, 0, Gl.GL\_RGBA, w, h, 0, Gl.GL\_RGBA, Gl.GL\_UNSIGNED\_BYTE, pixels);

break;

}

return texObject;

}

private void loadImageToolStripMenuItem\_Click\_1(object sender, EventArgs e)

{

DialogResult res = openFileDialog1.ShowDialog();

if (res == DialogResult.OK)

{

Il.ilGenImages(1, out imageId);

Il.ilBindImage(imageId);

string url = openFileDialog1.FileName;

if (Il.ilLoadImage(url))

{

int width = Il.ilGetInteger(Il.IL\_IMAGE\_WIDTH);

int height = Il.ilGetInteger(Il.IL\_IMAGE\_HEIGHT);

int bitspp = Il.ilGetInteger(Il.IL\_IMAGE\_BITS\_PER\_PIXEL);

switch (bitspp)

{

case 24:

mGlTextureObject = MakeGlTexture(Gl.GL\_RGB, Il.ilGetData(), width, height);

break;

case 32:

mGlTextureObject = MakeGlTexture(Gl.GL\_RGBA, Il.ilGetData(), width, height);

break;

}

textureIsLoad = true;

Il.ilDeleteImages(1, ref imageId);

}

}

}

// загрузка изображения с именем студента

private void LoadPicture()

{

Il.ilGenImages(1, out myImageId);

Il.ilBindImage(myImageId);

string fileName = "img/pri-117.jpg";

string fullPath = Path.GetFullPath(fileName);

if (Il.ilLoadImage(fullPath))

{

int width = Il.ilGetInteger(Il.IL\_IMAGE\_WIDTH);

int height = Il.ilGetInteger(Il.IL\_IMAGE\_HEIGHT);

int bitspp = Il.ilGetInteger(Il.IL\_IMAGE\_BITS\_PER\_PIXEL);

switch (bitspp)

{

case 24:

mGlTextureMyObject = MakeGlTexture(Gl.GL\_RGB, Il.ilGetData(), width, height);

break;

case 32:

mGlTextureMyObject = MakeGlTexture(Gl.GL\_RGBA, Il.ilGetData(), width, height);

break;

}

Il.ilDeleteImages(1, ref myImageId);

}

}

// отрисовка изображения с именем студента

private void DrawIndividualTask()

{

Gl.glBindTexture(Gl.GL\_TEXTURE\_2D, mGlTextureMyObject);

Gl.glRotated(180, 0, 1, 0);

Gl.glBegin(Gl.GL\_QUADS);

Gl.glVertex3d(1, 1, 0);

Gl.glTexCoord2f(0, 0);

Gl.glVertex3d(1, 0, 0);

Gl.glTexCoord2f(1, 0);

Gl.glVertex3d(0, 0, 0);

Gl.glTexCoord2f(1, 1);

Gl.glVertex3d(0, 1, 0);

Gl.glTexCoord2f(0, 1);

Gl.glEnd();

}

}

}

Результат выполнения

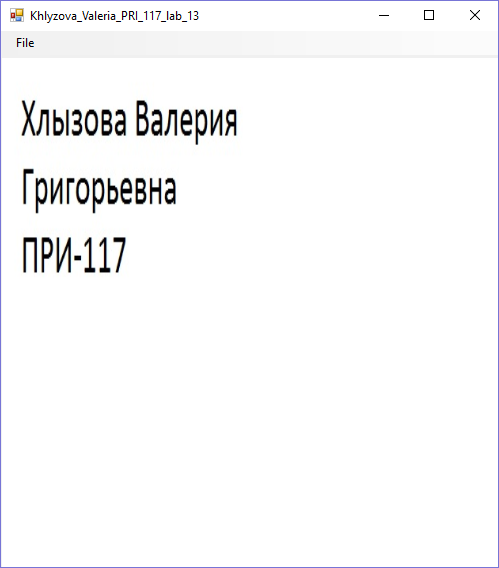


Рисунок 2 - Вывод информации о студенте.

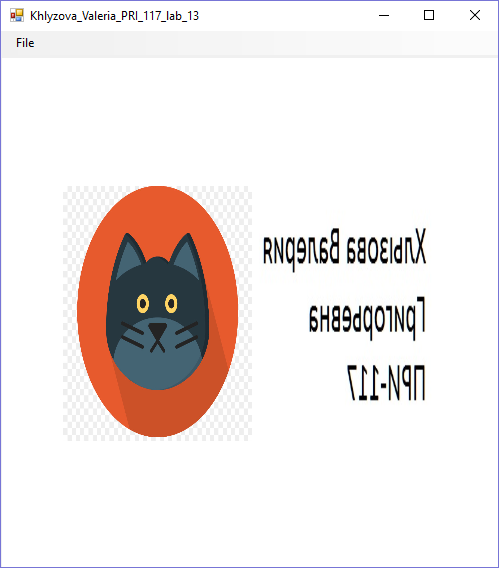
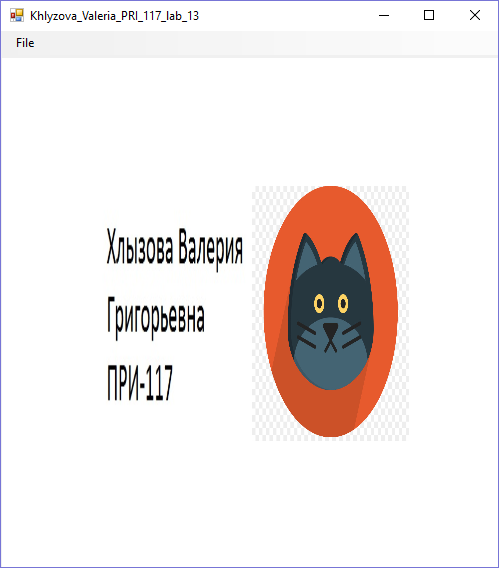


Рисунок 3 - Вращающиеся плоскости с изображением текстуры.

ВЫВОД

В ходе выполнения лабораторной работы были изучены способы реализации стилей заполнения поверхностей и закрашивания, которые имитируют сложную рельефную объемную поверхность, выполненную из определенного материала.