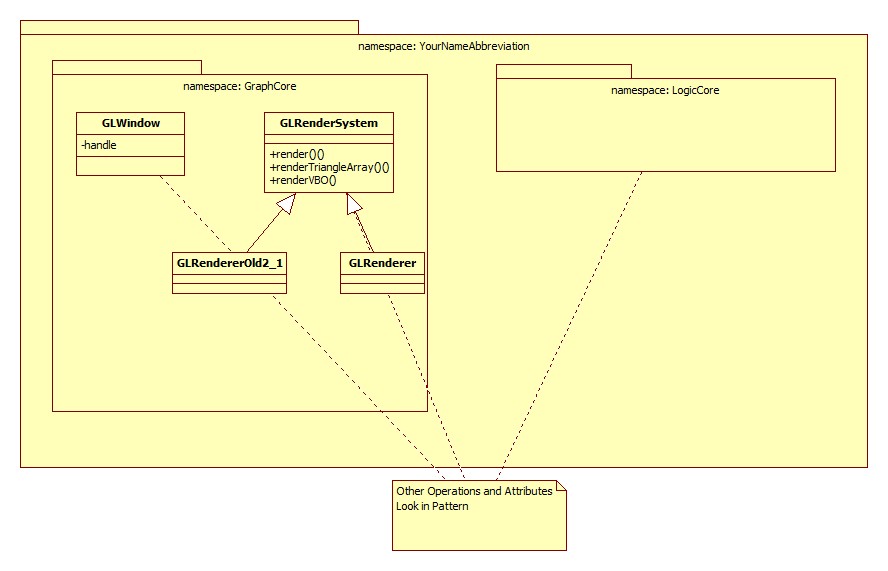
Лабораторна робота 2  
студента Москаленка Сергія, ПА-19-2

1. Створіть своє унікальне ім'я графічного рушія OpenGL - як абревіатуру вашого імені. Наприклад GSC - Григорович Сергій Константинович [(GSC Game World)](https://en.wikipedia.org/wiki/GSC_Game_World)

namespace GSC{

}

1. Використовуючи шаблон програми Task02Src [<https://github.com/KnightDanila/GraphicProjects_OpenGL_Shaders_GLSL/tree/master/Lesson2/Task02Src>] - запишіть та запустіть код з вашим простором імен.
2. Використовуйте діаграму UML для довідки: [](https://github.com/KnightDanila/GraphicProjects_OpenGL_Shaders_GLSL/blob/master/Lesson2/data/UML_Main1.jpg)
3. Додайте вихід до консолі:

Завдання 2

Автор: Москаленко Сергій

1. Створіть 3 вікна за допомогою класу GLWindow
2. Намалюйте об'єкт VBO (без текстури)
3. Намалюйте простий масив точок glDrawArrays()
4. Покажіть, що ви можете використовувати дві рендерні машини GLRender() та GLRendererOld2\_1().

int main()

{

//if (!glfwInit())return -1;

int w = 500, h = 500;

std::cout << "Завдання 2\nАвтор: Москаленко Сергій\n";

//if (w > h)

// glScalef((float)h / (float)w, 1.0f, 1.0f);

//else if (h > w)

// glScalef(1.0f, (float)w / (float)h, 1.0f);

bool is33 = false;

GraphCore::GLRenderSystem\* renderer1,\*renderer2;

if (is33)

{

renderer1 = new GLRender();

renderer2 = new GLRender();

}

else

{

renderer1 = new GLRendererOld2\_1();

renderer2 = new GLRendererOld2\_1();

}

renderer1->init();

glewExperimental = true; // Флаг необходим в Core-режиме OpenGL

GLWindow \*window1 = new GLWindow("first",w,h,0,30);

if (glewInit() != GLEW\_OK) {

fprintf(stderr, "Невозможно инициализировать GLEW\n");

return -1;

}

glfwMakeContextCurrent(window1->getGLFWHandle());

window1->setOpacity(0.8);

GLWindow \*window2 = new GLWindow("second",w,h,w,30);

glfwMakeContextCurrent(window2->getGLFWHandle());

window2->setOpacity(0.8);

GLWindow\* window3 = new GLWindow("third", w, h,0, h+30);

glfwMakeContextCurrent(window3->getGLFWHandle());

window3->setOpacity(0.8);

renderer2->init();

//GLWindow \*window2 = new GLWindow("second",w,h,500,30);

while (!glfwWindowShouldClose(window1->getGLFWHandle())||!glfwWindowShouldClose(window2->getGLFWHandle()))

{

//if (glfwGetKey(window1->getGLFWHandle(), GLFW\_KEY\_F))

// window1->ChangeFullScreen();

//if (glfwGetKey(window2->getGLFWHandle(), GLFW\_KEY\_F))

// window2->ChangeFullScreen();

glfwMakeContextCurrent(window1->getGLFWHandle());

glfwGetFramebufferSize(window1->getGLFWHandle(), &w, &h);

glViewport(0, 0, w, h);

renderer1->render(window1->getGLFWHandle(),0);

glfwSwapBuffers(window1->getGLFWHandle());

glfwMakeContextCurrent(window2->getGLFWHandle());

glfwGetFramebufferSize(window2->getGLFWHandle(), &w, &h);

glViewport(0, 0, w, h);

renderer2->render(window2->getGLFWHandle(),1);

glfwSwapBuffers(window2->getGLFWHandle());

glfwMakeContextCurrent(window3->getGLFWHandle());

glfwGetFramebufferSize(window3->getGLFWHandle(), &w, &h);

glViewport(0, 0, w, h);

renderer2->renderTriangleArray(window3->getGLFWHandle());

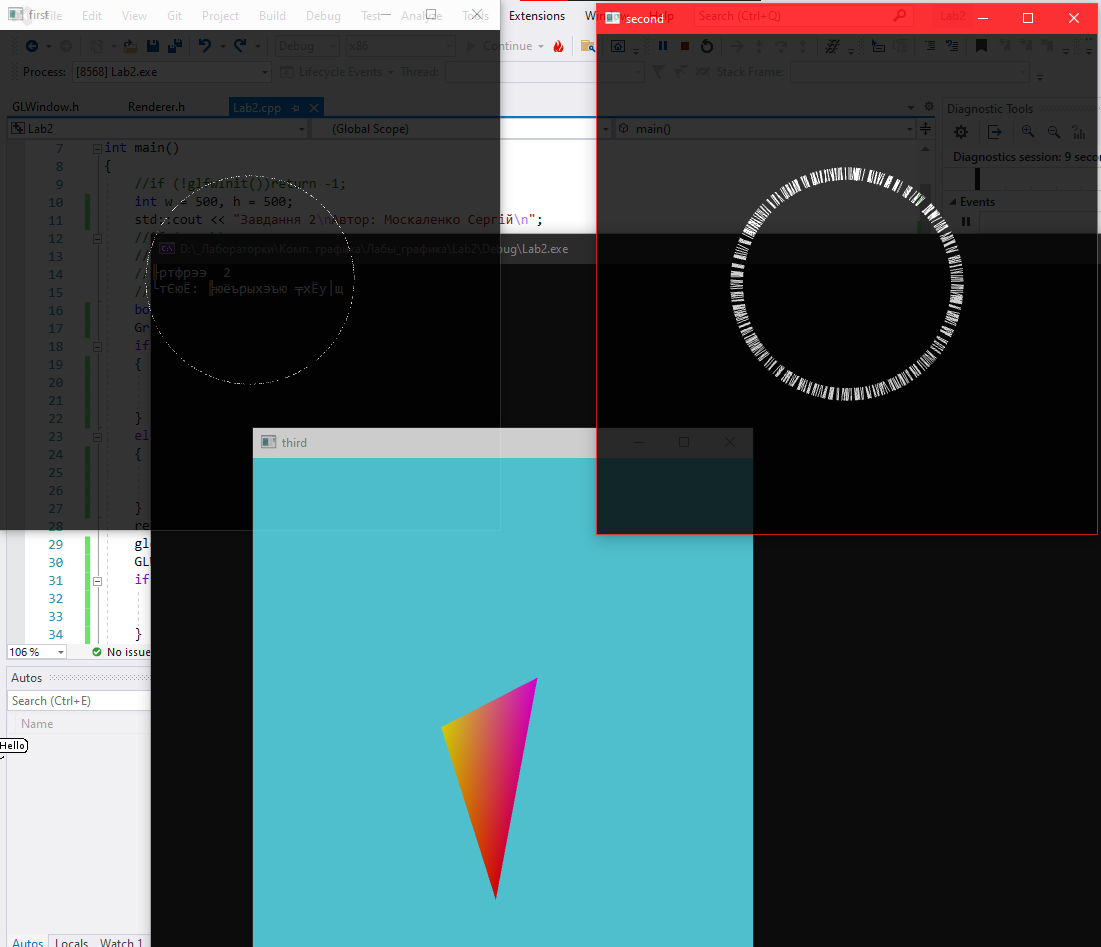
glfwSwapBuffers(window3->getGLFWHandle());

glfwPollEvents();

}

}

oldRenderer:



New renderer:

