Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего профессионального образования

«Владимирский государственный университет

имени Александра Григорьевича и Николая Григорьевича Столетовых»

(ВлГУ)

Кафедра информационных систем и программной инженерии

Лабораторная работа № 14

по дисциплине "Программирование графических приложений"

ТЕМА РАБОТЫ:

Анимация и пользовательский ввод

Выполнил:

студент гр. ПРИм-124

Парахин К.В.

Принял:

Жигалов И.Е.

Владимир 2024 г.

Цель работы:

Изучение методов анимации трехмерных объектов и способов управления объектами пользователем в WebGL.

Выполнение работы:

На основе приведенных примеров программ дополнить сцену, аналогичную построенной в результате выполнения лабораторной работы по теме 9:

- анимацией одного из объектов;

- перемещением одного их объектов мышкой;

- выводом информации об объекте при нажатии на его грани.

Применяемые анимация и управление должны соответствовать виду объектов

В лабораторной работе 9 была представлена сцена с рыбками. К ней можно применить в соответствии с заданием такие операции:

- одна из рыбок будет все время вращаться (как в примере 1)  
- другую рыбку можно будет перемещать к себе мышкой (как объекты в примере 4)  
- при нажатии на объект рыбки будет появляться на экране запись (как в примере 9)

Листинг кода:

        function animate() {

            sphere1.rotation.y += 0.02;

            renderer.render(scene, camera);

            requestAnimationFrame(animate);

        }

        function onMouseMove(event) {

            if (selectedObject) {

                mouse.x = (event.clientX / window.innerWidth) \* 2 - 1;

                mouse.y = -(event.clientY / window.innerHeight) \* 2 + 1;

                var newX = event.clientX - screenWidthHalf;

                var newY = -event.clientY + screenHeightHalf;

                // Ограничение движения объектов в пределах половины экрана

                newX = Math.max(-screenWidthHalf, Math.min(screenWidthHalf, newX));

                newY = Math.max(-screenHeightHalf, Math.min(screenHeightHalf, newY));

                selectedObject.position.set(newX / screenWidthHalf \* 75, newY / screenHeightHalf \* 75, 0);

            }

        }

        function onMouseDown(event) {

            mouse.x = (event.clientX / window.innerWidth) \* 2 - 1;

            mouse.y = -(event.clientY / window.innerHeight) \* 2 + 1;

            raycaster.setFromCamera(mouse, camera);

            var intersects = raycaster.intersectObjects([sphere1, sphere2, pyramid1, pyramid2]);

            if (intersects.length > 0) {

                selectedObject = intersects[0].object;

                infoBox.textContent = "Вы нажали на ребра объектов " + ++increment + " раз(а)";

            }

        }

    <div id="info">Зарегистировано 0 нажатий на объекты</div>

    <script>

        var scene, camera, renderer, pyramid1, sphere1, pyramid2, sphere2;

        var raycaster = new THREE.Raycaster();

        var mouse = new THREE.Vector2();

        var selectedObject = null;

        var infoBox = document.getElementById('info');

        var screenWidthHalf = window.innerWidth / 2;

        var screenHeightHalf = window.innerHeight / 2;

        var increment = 0;

        init();

        animate();

        function init() {

            renderer = new THREE.WebGLRenderer({ antialias: true });

            renderer.setSize(window.innerWidth, window.innerHeight);

            document.body.appendChild(renderer.domElement);

            camera = new THREE.PerspectiveCamera(70, window.innerWidth / window.innerHeight, 1, 1000);

            camera.position.z = 150;

            scene = new THREE.Scene();

            var obj\_material1 = new THREE.MeshPhongMaterial({ color: 0xff0000, side: THREE.DoubleSide });

            var obj\_material2 = new THREE.MeshPhongMaterial({ color: 0xaacb00, side: THREE.DoubleSide });

            var pyramid\_geometry = new THREE.CylinderGeometry(0, 16, 20, 3);

            var sphere\_geometry = new THREE.SphereGeometry(20, 6, 8);

            pyramid1 = new THREE.Mesh(pyramid\_geometry, obj\_material1);

            pyramid1.position.set(-40, 60, 0);

            pyramid1.rotation.y = -Math.PI / 2;

            sphere1 = new THREE.Mesh(sphere\_geometry, obj\_material1);

            sphere1.position.set(-10, 60, 0);

            pyramid2 = new THREE.Mesh(pyramid\_geometry, obj\_material2);

            pyramid2.position.set(-40, 0, 0);

            pyramid2.rotation.y = -Math.PI / 2;

            sphere2 = new THREE.Mesh(sphere\_geometry, obj\_material2);

            sphere2.position.set(-10, 0, 0);

            var light = new THREE.HemisphereLight(0x666666, 0xff0000, 1);

            scene.add(light);

            scene.add(pyramid1, sphere1, pyramid2, sphere2);

            renderer.domElement.addEventListener('mousedown', onMouseDown, false);

            renderer.domElement.addEventListener('mousemove', onMouseMove, false);

            renderer.domElement.addEventListener('mouseup', onMouseUp, false);

            window.addEventListener('resize', onWindowResize, false);

        }

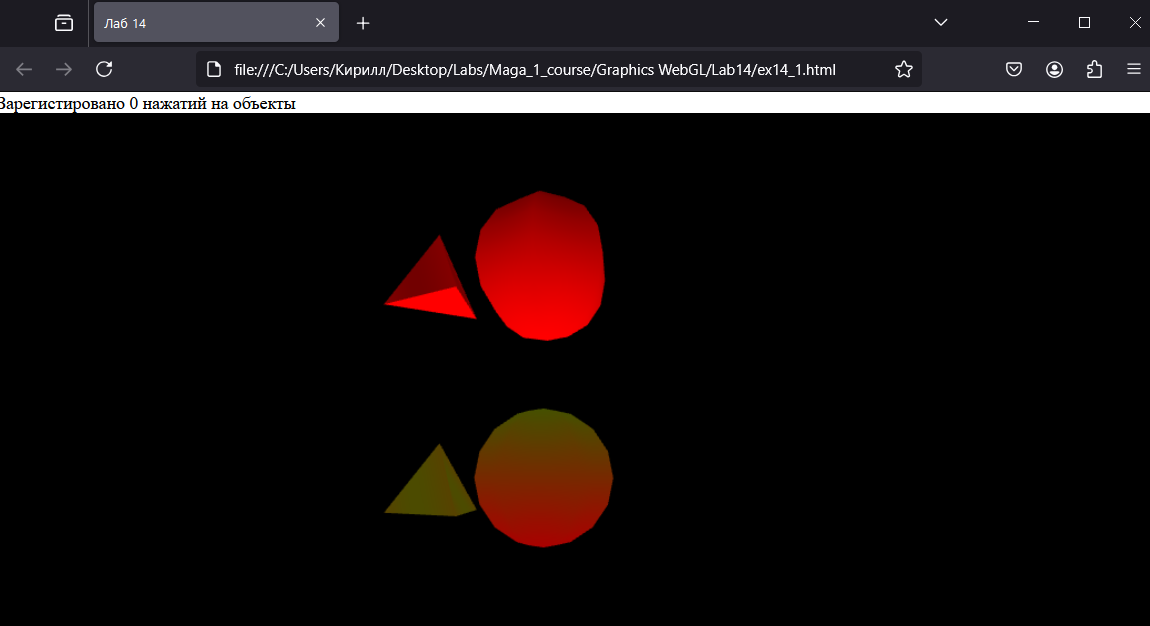


Рисунок 1. Страница при загрузке

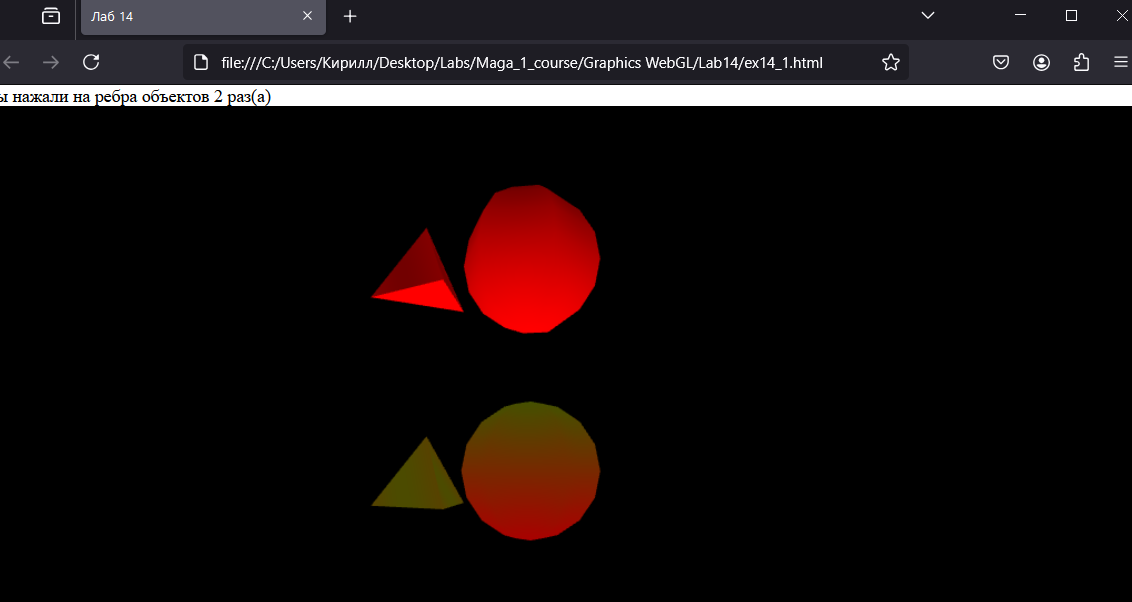


Рисунок 2. После вращения и нескольких нажатий

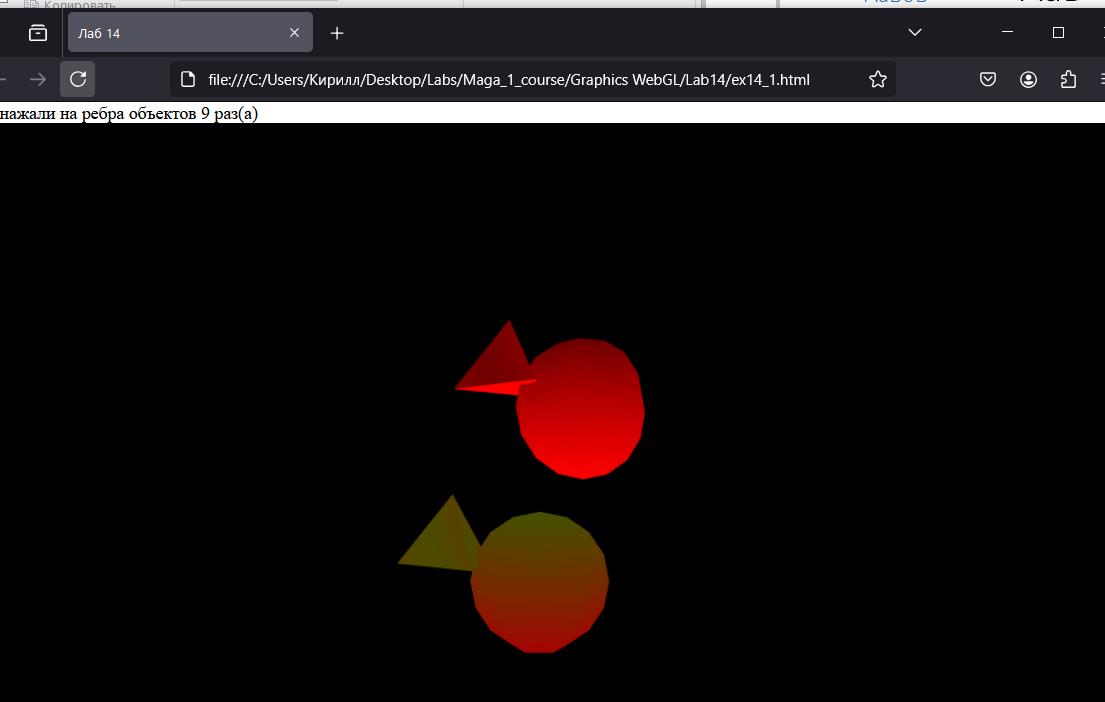


Рисунок 3. После перемещения объектов

Вывод

В результате выполнения работы я провел изучение методов анимации трехмерных объектов и способов управления объектами пользователем в WebGL.