

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ

Волжский политехнический институт (филиал)
федерального государственного бюджетного образовательного учреждения
высшего образования
«Волгоградский государственный технический университет»
(ВПИ (филиал) ВолгГТУ)

Факультет Инженерно-экономический факультет
Кафедра ВИТ

Лабораторная работа №7

по дисциплине: Компьютерная графика

на тему: OpenGL: визуализация трехмерных объектов с
использованием изображений и текстур.

Вариант № 17

Студент Семёнов Андрей Игоревич
(фамилия, имя, отчество)

Группа ВИП-308

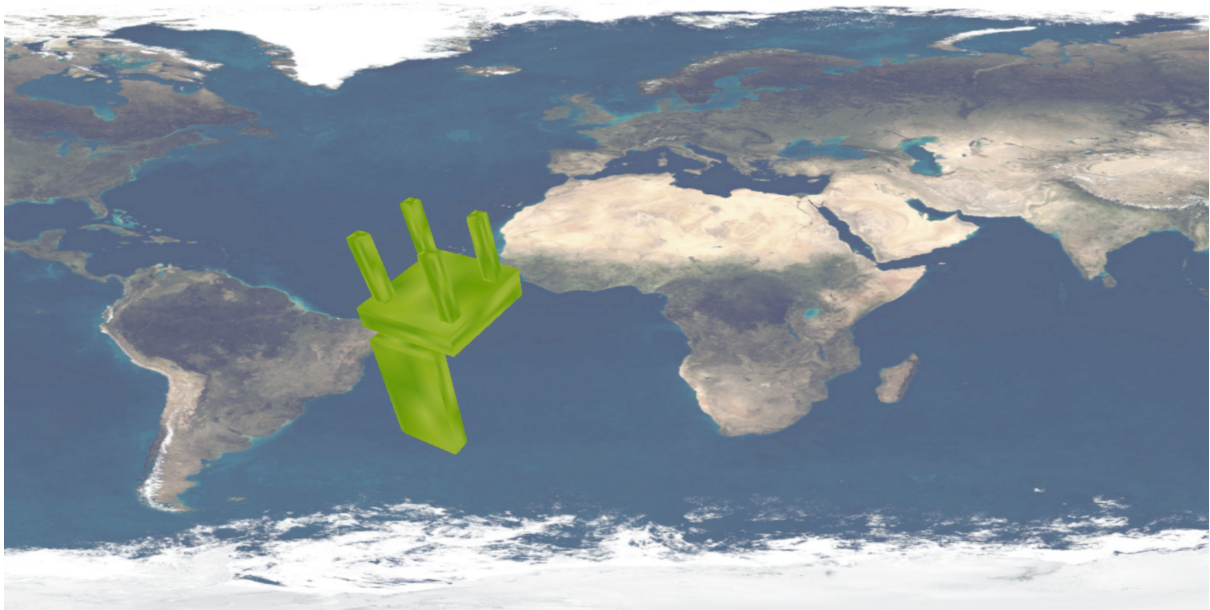
Оценка _____
(баллы)

Проверил _____
(подпись и дата подписания)

доцент, Абрамова О.Ф.
(долж., фамилия и инициалы)

Волжский, 2023

Скриншот работы:



Код:

```
import * as THREE from "three";
import { GLTFLoader } from "three/addons/loaders/GLTFLoader.js";

const scene = new THREE.Scene();
const camera = new THREE.PerspectiveCamera(
  75,
  window.innerWidth / window.innerHeight,
  0.1,
  1000
);

const loader = new GLTFLoader();
const textureLoader = new THREE.TextureLoader();

let chair;

loader.load(
  "/chair.glb",
  function (gltf) {
    chair = gltf.scene.children[0];
    textureLoader.load("/grass.png", (texture) => {
      const material = new THREE.MeshBasicMaterial({ map: texture });
      chair.material = material;
      scene.add(chair);
    });
  },
```

```

    undefined,
    function (error) {
        console.error(error);
    }
);

textureLoader.load("/earth_atmos_2048.jpg", (texture) => {
    const geometry = new THREE.PlaneGeometry(10, 10, 1);
    const material = new THREE.MeshBasicMaterial({ map: texture });
    const backgroundMesh = new THREE.Mesh(geometry, material);

    backgroundMesh.position.z = -5;
    const aspectRatio = texture.image.width / texture.image.height;
    backgroundMesh.scale.x = camera.aspect * aspectRatio;
    backgroundMesh.scale.y = 1;

    scene.add(backgroundMesh);
});

const renderer = new THREE.WebGLRenderer();
renderer.setSize(window.innerWidth, window.innerHeight);
document.body.appendChild(renderer.domElement);

camera.position.z = 2;
camera.position.y = 0.5;

setInterval(() => {
    if (chair) {
        chair.position.x = -3.5;
    }
}, 6000);

function animate() {
    requestAnimationFrame(animate);

    if (chair) {
        chair.position.x += 0.02;
        chair.rotation.y += 0.02;
        chair.rotation.x += 0.02;
    }

    renderer.render(scene, camera);
}

animate();

```