

МИНОБРНАУКИ РОССИИ

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ ИМ. Р.Е. АЛЕКСЕЕВА»
(НГТУ)**

Институт радиоэлектроники и информационных технологий
Кафедра «Прикладная математика»

Учебная дисциплина «Компьютерная графика»

Лабораторная работа №6

Выполнил:

Козловская Анна Сергеевна, 19-ПМ-2

Проверил:

Заведующий кафедрой «Графические информационные системы»

Филинских А.Д.

Нижний Новгород

2022

Содержание

Постановка задачи.....3

Решение задачи.....4

Вывод.....6

Постановка задачи

Используя язык Javascript и библиотеку Three.js, выполнить следующие задачи:

1. Добавить орбиту Земли.
2. Написать функцию для добавления орбиты по заданному радиусу.
3. Добавить орбиты к оставшимся планетам.

Решение задачи

Сначала создадим произвольную геометрию с помощью `THREE.Geometry()`. Затем создадим материал с помощью `THREE.Points.Material`. Затем в цикле пропишем координаты каждой точки и добавим эти точки к нашей геометрии. Затем создаем объект с помощью `THREE.Points` по нашей геометрии и материалу. В итоге получим орбиту из точек, на которую положим тень с помощью `shadowCast`. Полностью создание орбиты Земли выглядит так:

```
var geometry = new THREE.Geometry();
var material = new THREE.PointsMaterial({ color: 0xffffff, size: 1, sizeAttenuation: false });
for (var i = 0; i < 200000; i++) {
  var vertex = new THREE.Vector3();
  vertex.x = Math.sin(180 / Math.PI * i) * 7500;
  vertex.z = Math.cos(180 / Math.PI * i) * 7500;
  geometry.vertices.push(vertex);
}
earth_orbit = new THREE.Points(geometry, material);
earth_orbit.castShadow = true;
scene.add(earth_orbit);
```

Теперь напишем функцию для добавления орбиты, чтобы не прописать ее создание для каждой планеты. Создаем функцию `Orbit()`, которая принимает аргумент `radius`. Внутри `Orbit()` создаем вложенную функцию `draw()`, которая повторяет действия при создании орбиты Земли (только теперь при задании координат вектора используется `this.radius`, а не конкретное число). Далее в функции `draw()` на сцену добавляется созданная орбита.

Использование `Orbit`, например, для Меркурия, выглядит следующим образом:

```
var mercury_orbit = new Orbit(4000);
mercury_orbit.draw(scene);
```

Далее добавляем орбиты ко всем планетам тем же образом. И получаем следующую картинку:

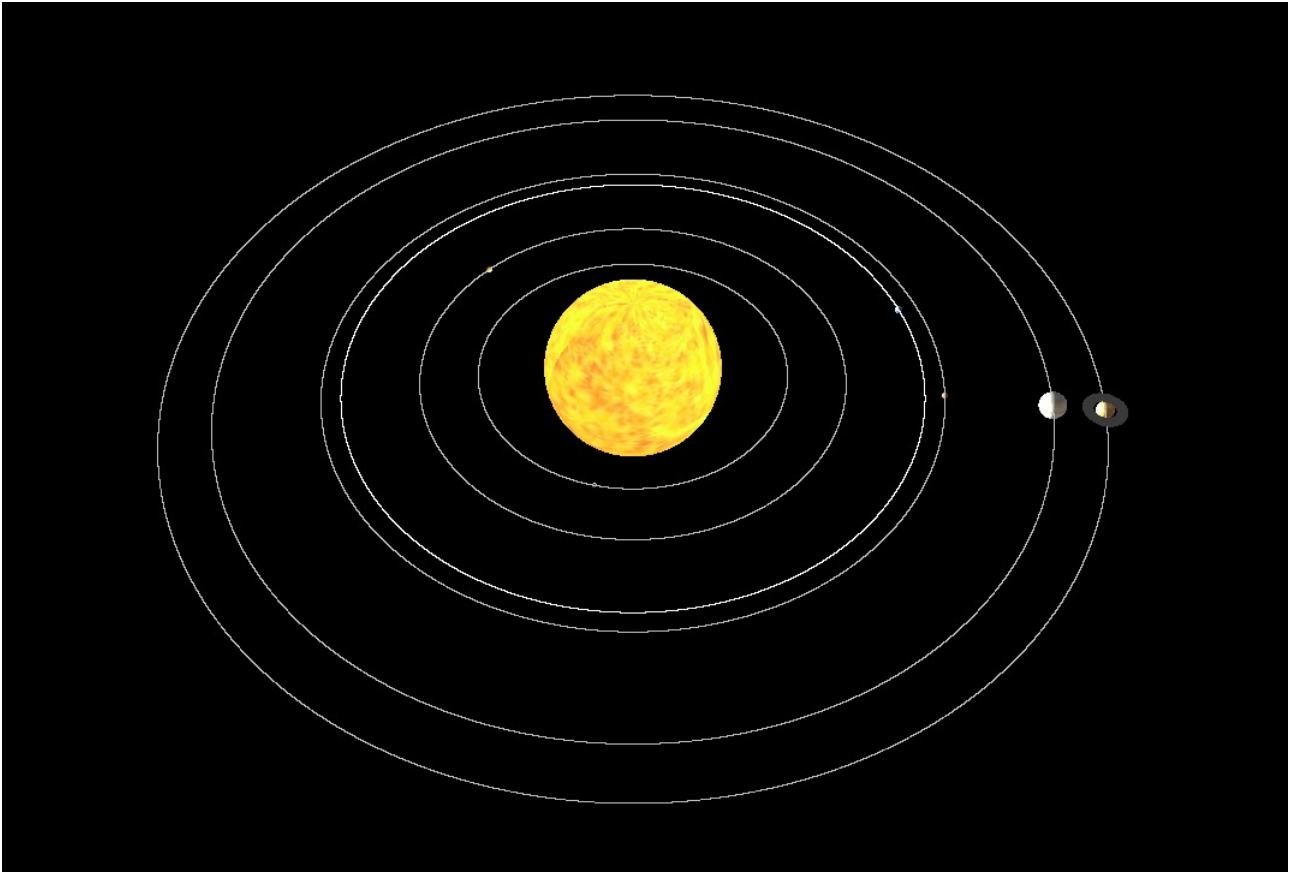


Рисунок 1 — Система с орбитами

Вывод

Таким образом, вручную была добавлена орбита Земли, после чего была написана функция для добавления орбиты по радиусу и были добавлены орбиты ко всем оставшимся планетам.