Физико-математический лицей № 239

**Моделирование движения небесных тел**

Работу выполнил

Ученик 10-2 класса

Ермошин Иван

Санкт-Петербург

2017-2018

1. Постановка задачи

Посчитать движение небесных тел по известным траекториям в трехмерном пространстве, запрограммировать их вывод на экран, добавить несколько дополнительных функций, таких как: добавление новых планет.

2. Уточнение исходных и выходных данных и ограничений на них.

Вводятся данные сразу в работающую версию программы или можно задать планеты в коде.

Основные параметры: цвет планеты/орбиты, размер планеты, эксцентриситет, малая полуось, скорость движения, углы наклона к осям.

Тип double.

3. Выбор метода решения

Создаю класс Body, включающий в себя цвет, размер, вращение тела вокруг своей оси и траекторию.

Под траекторию и вращение выделяю отдельные классы.

В траектории создаю методы для получения координат тела в следующий момент времени.

Передаю эти координаты в render (объект библиотеки THREE.js), он же отображает тела на экране.

4. Математическая модель.

Я использую некоторые очевидные формулы для эллипса: считаю координату от угла между прямой отсчета и направлением на тело и умножению на матрицу поворота чтобы получить ее же, но в нужной плоскости.

Можно привести доказательство, но оно столь очевидно, что есть даже в [википедии](https://ru.wikipedia.org/wiki/Матрица_поворота).

5.Анализ решения.

Я используя язык javascript, он удобен для ООП и там есть довольно удобная библиотека для рисования.

Для рисования я использую фреймворк THREE.js, в нём прописаны формы тел, кривых, это очень удобная вещь с подробной документацией на английском.

Можно было бы писать на другом языке, но там сразу резко встает вопрос о графическом редакторе: их много и они не всегда просты для изучения.

THREE.js же наоборот доступен для использования и изучения.

Еще можно проанализировать мою мат.модель: у меня планеты летают по готовым эллипсам, можно было задать движение планет по ОТО или закону тяготения Ньютона, но ОТО это сложно, а Ньютон выдавал бы погрешность и система не была бы стабильной: планеты слетались бы в центр если достаточно долго подождать, для стабильности можно численно решить диффуры методом Рунге-Куты, но это опять же сложно с точки зрения математики.

Тем более, что солнечная система довольно стабильна (как минимум посчитаны параметры планет и они не очень-то меняются со временем, если бы это было не так, то данные в википедии хотя раз в месяц, но менялись): если вбить параметры планет солнечной системы, то программа будет достаточно точно моделировать ее эволюцию.