# МИНОБРНАУКИ РОССИИ

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

# «ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)

**Кафедра МО ЭВМ**

# ОТЧЕТ

**по лабораторной работе**

# по дисциплине «Вычислительная математика»

**Тема: Уравнение движения заряженной частицы в электромагнитном поле.**

Студенты гр. 0382 Диденко Д.В.

Кривенцова Л.С.

Преподаватель Павлов Д.А.

Санкт-Петербург 2021

# Цель работы.

# Исследовать дифференциальное уравнение движения заряженной частицы в электромагнитном поле с помощью программы на языке программирования.

# Задание.

Решить систему дифференциальных уравнений аналитическим методом. Написать программу, которая применяет различные методы вычислений и реализовать вывод значений в файл, а также в виде графиков. Сравнить результаты, полученные с помощью вычислений различными способами.

**Основные теоретические положения.**

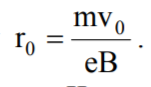
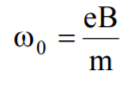
Система дифференциальный уравнений.

Системой дифференциальных уравнений называется совокупность уравнений, в каждое из которых входят независимая переменная, искомые функции и их производные. Всегда предполагается, что число уравнений равно числу неизвестных функций.

Движение заряженных частиц в электромагнитных полях.

Движение заряженной частицы в магнитном поле описывается вторым законом Ньютона:

ma = F, (1)

где m - масса частицы, F = e[vB] - сила Лоренца (В - вектор магнитной индукции, е - заряд частицы). Наиболее простой вид имеет движение заряженной частицы в однородном магнитном поле, т. е. в поле, направление и величина которого одинаковы во всех точках пространства. Заряженная частица движется по поверхности воображаемого цилиндра с радиусом, равным ларморовскому радиусу  Ось цилиндра направлена вдоль вектора индукции магнитного поля. Частица вращается вокруг оси цилиндра с постоянной угловой скоростью, которая называется циклотронной частотой.

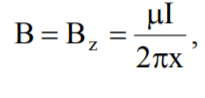
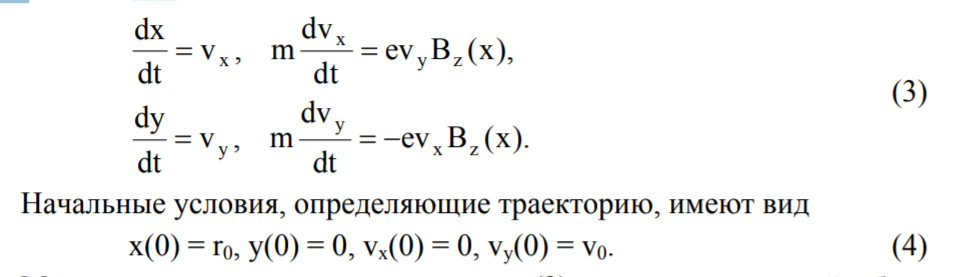
Зависимость напряженности магнитного поля от пространственных координат резко усложняет решение уравнений движения.

Движение заряженной частицы в поле прямолинейного тока.

Пусть в магнитном поле бесконечного прямолинейного проводника, по которому течет ток I, движется частица с положительным зарядом е и массой m. Начальная скорость частицы vо направлена вдоль тока в проводнике. В начальный момент времени частица расположена относительно проводника на расстоянии, равном локальному значению ларморовского радиуса, рассчитанного по величине магнитного поля в точке расположения частицы:

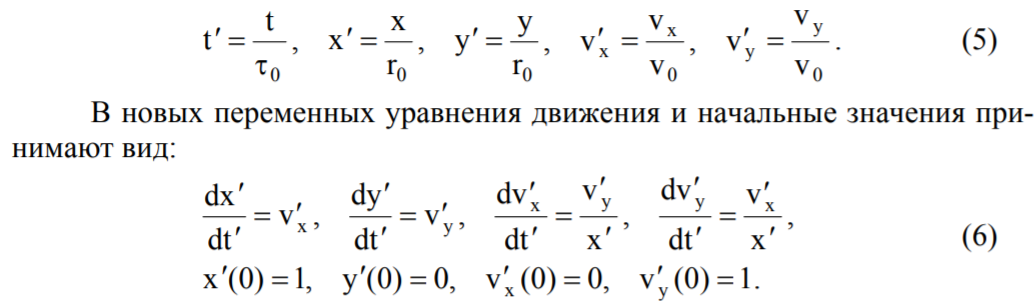


Рассмотрим плоскость, проходящую через проводник и частицу, и

введем декартовы координаты с осью X, проходящей через частицу перпендикулярно проводнику, осью Y с положительным направлением вдоль тока и осью Z, перпендикулярной плоскости. В каждой точке плоскости магнитное поле линейного проводника перпендикулярно этой плоскости и равно  и сила Лоренца не имеет составляющей вдоль оси Z. Поэтому если частица имела в начальный момент времени равную нулю z компоненту скорости, то ее траектория во все последующие моменты времени будет лежать и рассматриваемой плоскости. Уравнения движения в этом случае имеют вид

Математически уравнения движения (3) являются системой обыкновенных дифференциальных уравнений первого порядка для четырех функций: х(t), у(t), vx(t) и vу(t). Начальные условия (4) определяют задачу Коши для этой системы уравнений.

При выполнении расчетов часто неудобно использовать размерные значения величин. В рассматриваемой задаче характерным временем, за которое происходят типичные изменения траектории, является величина τ0, обратная циклотронной частоте ω, рассчитанной по величине индукции магнитного поля в какой-либо точке траектории (например, в начальной). Характерным размером траектории частицы является значение ларморовского радиуса r0, рассчитанного по магнитной индукции и скорости в этой же точке. Характерные размерные величины связаны соотношением r0 = v0⋅τ0. Для решения задачи целесообразно ввести следующие безразмерные переменные:



# Выполнение работы.

**Ход решения:**

Краткое описание методов, которые мы применяем

Для численного решения имеющийся системы дифференциальных уравнений были реализованы следующие методы:

* Метод Эйлера-Коши(Метод Хьюна).
* Метод Эйлера.
* Аналитический метод.

1. Метод Эйлера-Коши(Метод Хьюна).

Принцип: (формулы, шаги)

Пояснения с кодом: (реализация)

2. Метод Эйлера.

Принцип: (формулы, шаги)

Пояснения с кодом: (реализация)

3. Аналитический метод.

Принцип: (формулы, шаги)

Пояснения с кодом: (реализация)

4. Сравнить показатели.

(Ну что они примерно равны, вычисления верные и блаблабла)

5. Графический интерфейс.

Для графического представления полученных с помощью вышеуказанных методов результатов используется мультимедийная библиотека SFML, а точнее её графический модуль (*SFML/Graphics.hpp*). Вся работа с применением данной библиотеки выделена в отдельный класс *DrawLines*. Этот класс не умеет полей, в нём реализован лишь метод отрисовки графиков - *void prin*t с модификатором доступа *public*.

Этот метод принимает на вход аргументы:

*int N*; - количество вычисленных точек.

*float \*\*array\_x, float \*\*array\_y, float \*\*array\_z* – двумерные массивы, содержащие координаты заряженной частицы, по которым и будут строиться графики.

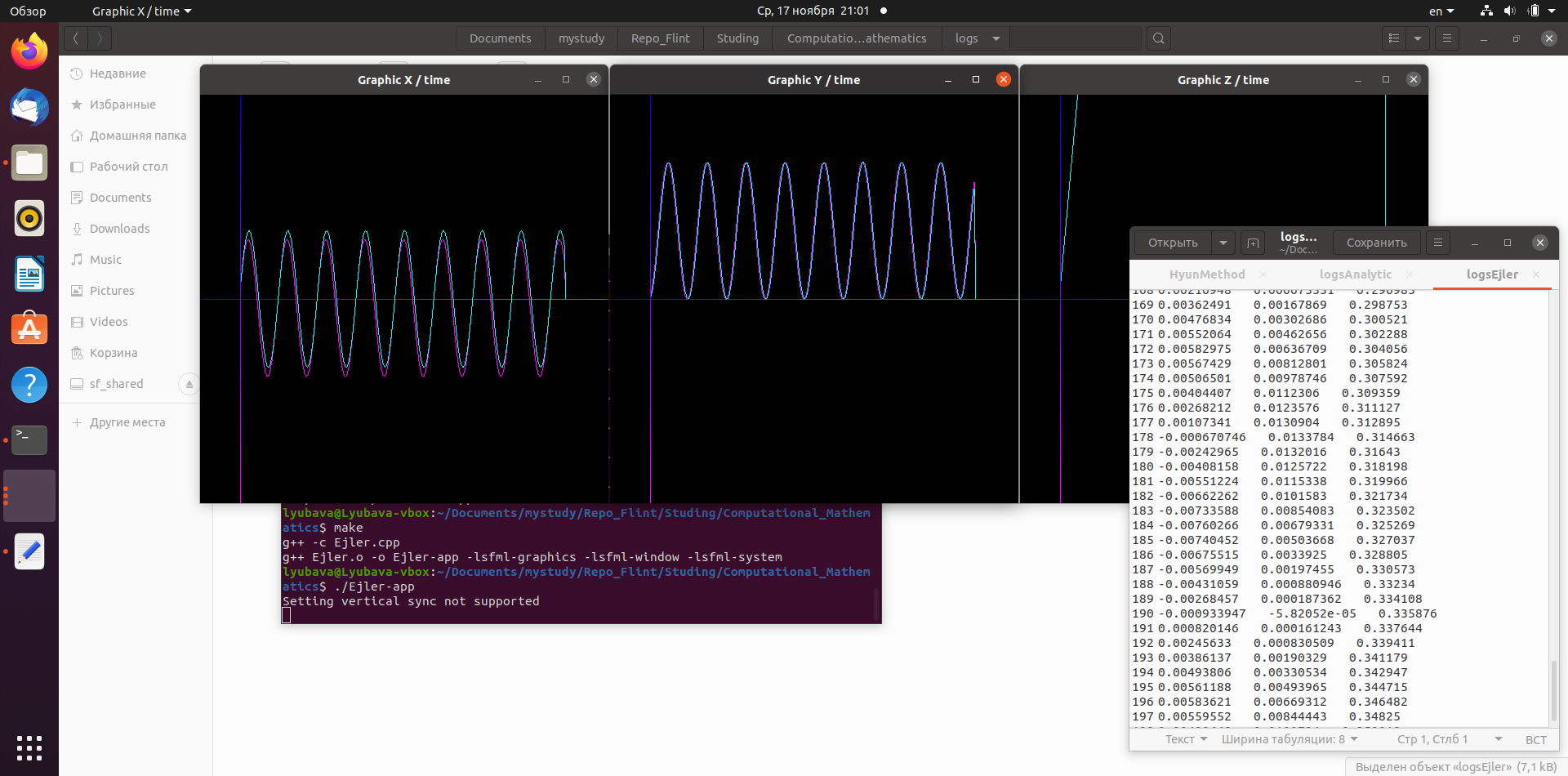
В методе мы сначала задаём размер окна. После этого создаём три объекта окна (*windowX*, *windowY*, *windowZ*) как экземпляры класса *RenderWindow*. С помощью класса классом *VertexArray* создаём две линии – они будут служить координатными осями. Эти линии в конце метода будут отрисованы в каждом окне. Пользуясь полями *position* и *color* задаём положение и цвет точек.

Используя тот же класс, создаём ещё несколько линий: соединённые точки, координаты которых представляют из себя время движения частицы и её нахождение по оси X (полученные тремя разными методами) – сохраняются в три массива (три линии), аналогично для нахождения частицы по оси Y и Z.

Все точки сохраняются в массиве с учётом их печати на экран: значение координаты инвертируется (т.к. начало отсчёта находится вверху графического окна), также к положению точки прибавляются координаты центра системы координат.

Далее создаётся конструктор обработчика событий, в котором отслеживается закрытие окон *(event.type == Event::Closed*) и производятся вызовы метода *draw*, которое рисует в окнах созданные ранее массивы соединённых между собой точек – траектории движения заряженной частицы по осям пространства. Таким образом на экране мы видим три окна, в каждом из которых нарисованы оси координат и графики вычисленных значений.

**Результат работы программы:**

Рис 1. – демонстрация работы программы в терминалe Ubuntu.

Программа открывает три окна с графическим интерфейсом: каждое из них печатает траектории заряженной частицы по определенной координате (значения X, Y или Z, зависимых от времени), полученные в результате реализации методов численного интегрирования. Все эти значения, помимо графиков, программа записывает в текстовые файлы (отдельный файл для каждого метода).

Обозначения графиков:

Горизонтальная ось – прошедшее время с начала движения;

Вертикальная ось – значения координаты заряженной частицы по оси (X, Y или Z).

График первого метода рисуется белым цветом.

График второго метода рисуется фиолетовым цветом.

График третьего метода рисуется голубым цветом.

Нумерация метода – порядок его вызова в программе.

# Выводы.

Были изучены и реализованы некоторые методы численного интегрирования, с помощью которых было реализовано вычисление системы дифференциальных уравнений движения заряженной частицы в электромагнитном поле. Для наглядности результатов вычислений был подключен графический интерфейс (рисование графиков).

Кроме того, был проведен сравнительный анализ, который подтвердил результаты, полученные в ходе работы программы.