



ФАКУЛЬТЕТ КОМПЬЮТЕРНЫХ НАУК И ТЕЛЕКОММУНИКАЦИЙ

Лабораторная работа №4
По дисциплине
«Численные методы и прикладное программирование»

Тема:
«Численные методы решения дифференциальных уравнений
первого порядка»

Работу выполнили:

*Дзенис Ричард
Кобелев Денис
Якушин Владислав*

Содержание

1	Формулировка задания	2
2	Метод Эйлера	2
2.1	Результаты работы метода	2
3	Метод Рунге-Кутты 4-ого порядка	2
3.1	Результаты работы метода	2
4	Выводы	3

1 Формулировка задания

В данной лабораторной работе требуется реализовать два метода решения дифференциальных уравнений: метод Эйлера и индивидуальный метод (метод Рунге-Кутты 4-ого порядка). Само дифференциальное уравнение выдается преподавателем во время проведения лабораторных работ. Для того чтобы реализовать алгоритмы, требуется предусмотреть ряд входных параметров.

2 Метод Эйлера

```
1 #include "common.h"
2
3 std::vector<point> solve(
4     FTY f, double y0,
5     double left, double right,
6     double step)
7 {
8     std::vector<point> points { {left, y0} };
9     for (double t = left + step; t < right + step; t += step) {
10         auto y_k = points.back().y;
11         points.emplace_back(point{ t, y_k + step * f(t, y_k) });
12     }
13     return points;
14 }
15
16 #include "main.cpp"
```

2.1 Результаты работы метода

3 Метод Рунге-Кутты 4-ого порядка

```
1 #include "common.h"
2
3 std::vector<point> solve(
4     FTY f, double y0,
5     double left, double right,
6     double step)
7 {
8     std::vector<point> points { {left, y0} };
9     for (double t = left + step; t < right + step; t += step) {
10         auto [t_k, y_k] = points.back();
11         auto f_1 = f(t_k, y_k);
12         auto f_2 = f(t_k + step / 2.0, y_k + step / 2.0 * f_1);
13         auto f_3 = f(t_k + step / 2.0, y_k + step / 2.0 * f_2);
14         auto f_4 = f(t, y_k + step * f_3);
15         points.emplace_back(point{
16             t,
17             y_k + step / 6.0 * (f_1 + 2 * f_2 + 2 * f_3 + f_4)
18         });
19     }
20     return points;
21 }
22
23 #include "main.cpp"
```

3.1 Результаты работы метода

4 Выводы