Университет ИТМО

Факультет: ПИиКТ

Дисциплина: Вычислительная математика

Лабораторная работа №5

Вариант “2”

Выполнили: **Кудлаков Роман**

Группа: **P3231**

Преподаватель: **Перл Ольга Вячеславовна**

Санкт-Петербург, 2021

Описание метода

Усовершенствованный метод Эйлера – метод приближенного вычисления для нахождения решения ДУ вида $y' = f(x, y)$ с начальным условием (x_0, y_0) на некотором промежутке $(a,\ b)$. Идея данного метода заключается в том, чтобы заменить фрагмент графика $y = f(x)$ ломаной линией, где ломаные – это касательные, проведенные к серединам каждого из отрезка графика функции.

Алгоритм метода

1. Ввод ДУ, точки (x_0, y_0), через которую проходит график решения, и точность решения

2. Производится подсчет длины шага. Из-за того, что порядок точности метода равен 2, то длину шага можно вычислить по следующей формуле:

$$h = \sqrt{\varepsilon}$$

3. Теперь для каждого шага длины h, начиная с x_0, нужно выполнить следующие операции:

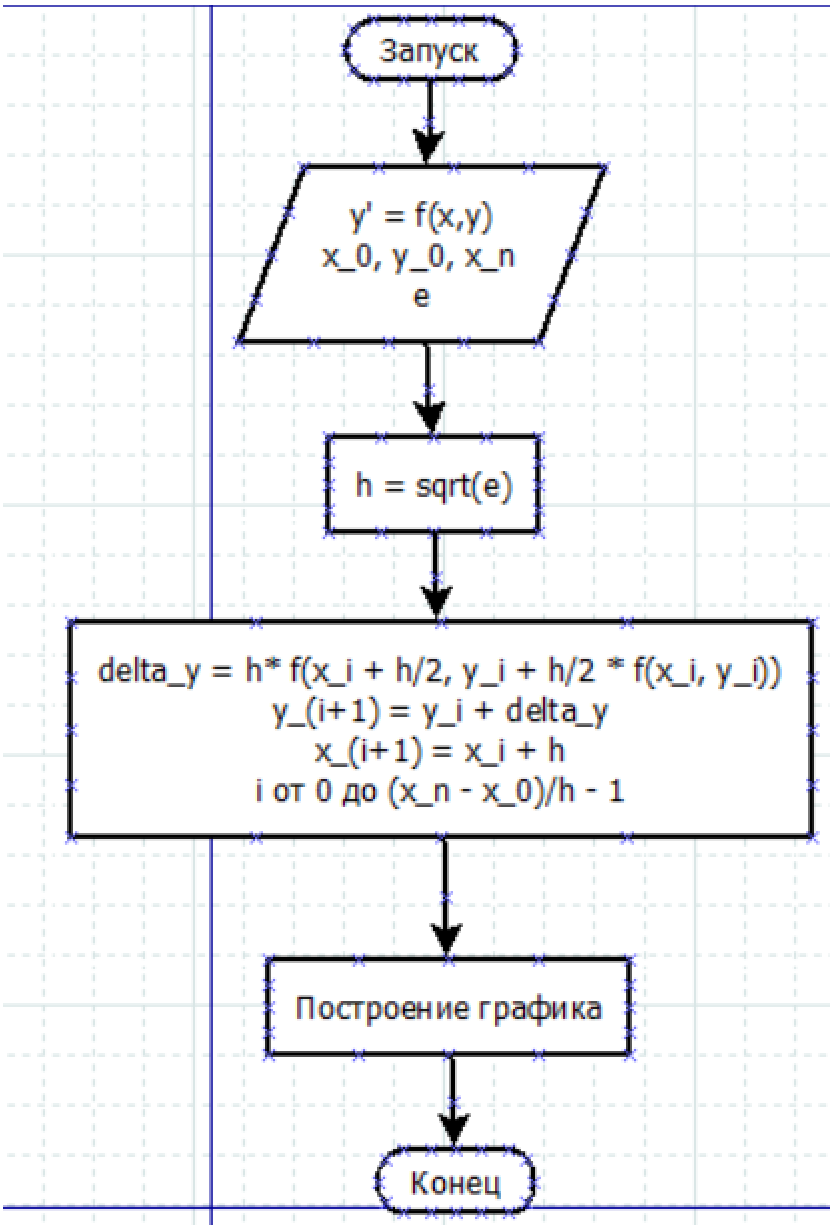
$$\Delta y = h * f\left(x_i + \frac{h}{2}, \qquad y_i + \frac{h}{2} * f(x_i, y_i)\right)$$
$$y_{i+1} = y_i + \Delta y$$
$$x_{i+1} = x_i + h$$

4. В итоге объединим все полученные пары точек (x_i, y_i) $i =\ 0, 1, \dots, (\frac{b-a}{h} - 1)$ и получим график решения ДУ

Листинг кода

```
def advanced_euler_method(func, x0, y0, xn, step):
    x = [x0]
    y = [y0]
    current_x = x0
    current_y = y0
    while current_x < xn:
        current_res_func = func.subs([('x', current_x), ('y', current_y)]).evalf()
        delta_y = step * func.subs([('x', current_x + step/2), ('y', current_y +
step/2 * current_res_func)]).evalf()
        current_y += delta_y
        current_x += step
        x.append(current_x)
        y.append(current_y)
    return x, y
```

Блок-схема

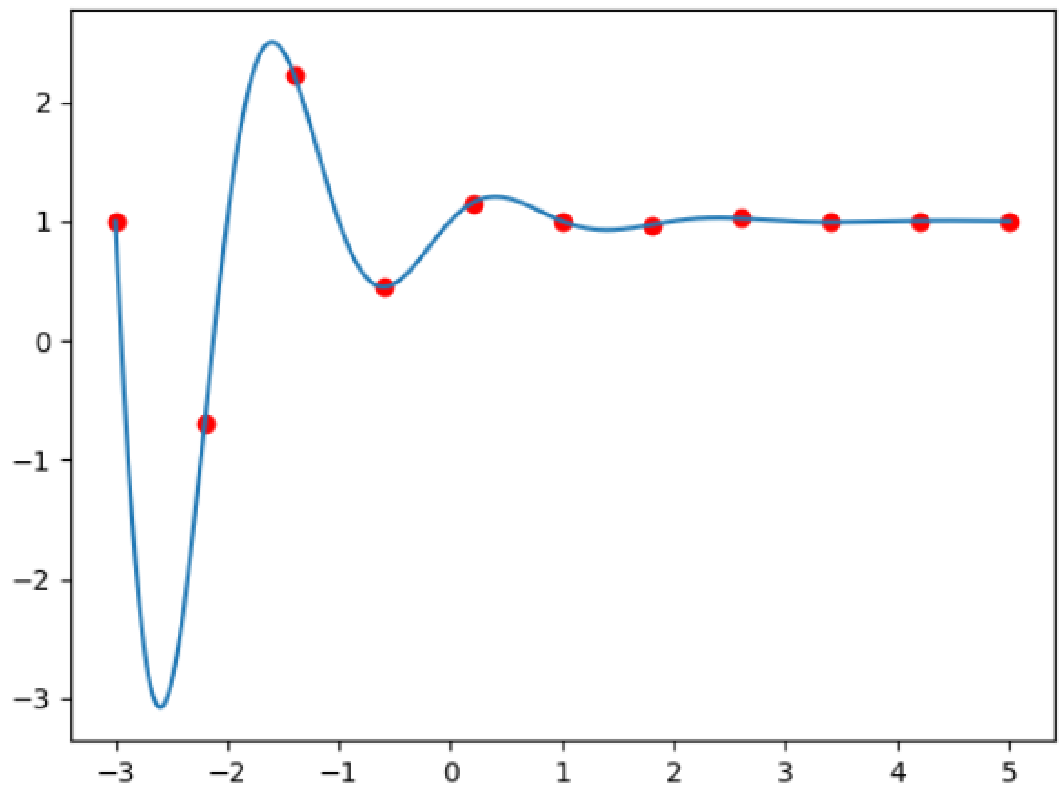


Тесты и результаты

Тест 1.

```
Choose one of differential equations:
1) y' - x^2 - 2y = 0
2) y' =  sin(x)
3) y' = 1 + e^(-x) * cos(pi * x) - y
3
Input start points
x = -3
y = 1
Input end point
5
Input precision
3
```

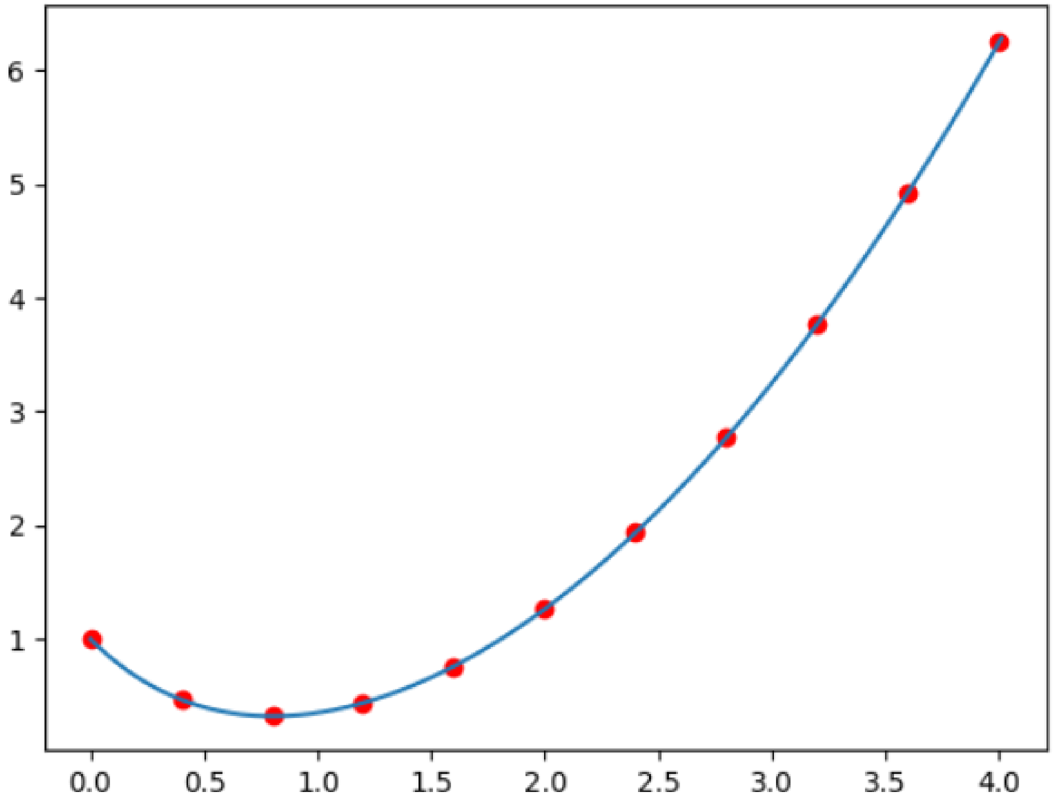
Точная функция: $y(x) = \left(e^{x} + \frac{\sin(\pi x)}{\pi}\right) e^{-x}$



Тест 2.

```
Choose one of differential equations:
1) y' - x^2 - 2y = 0
2) y' =  sin(x)
3) y' = 1 + e^(-x) * cos(pi * x) - y
1
Input start points
x = 0
y = 1
Input end point
4
Input precision
4
```

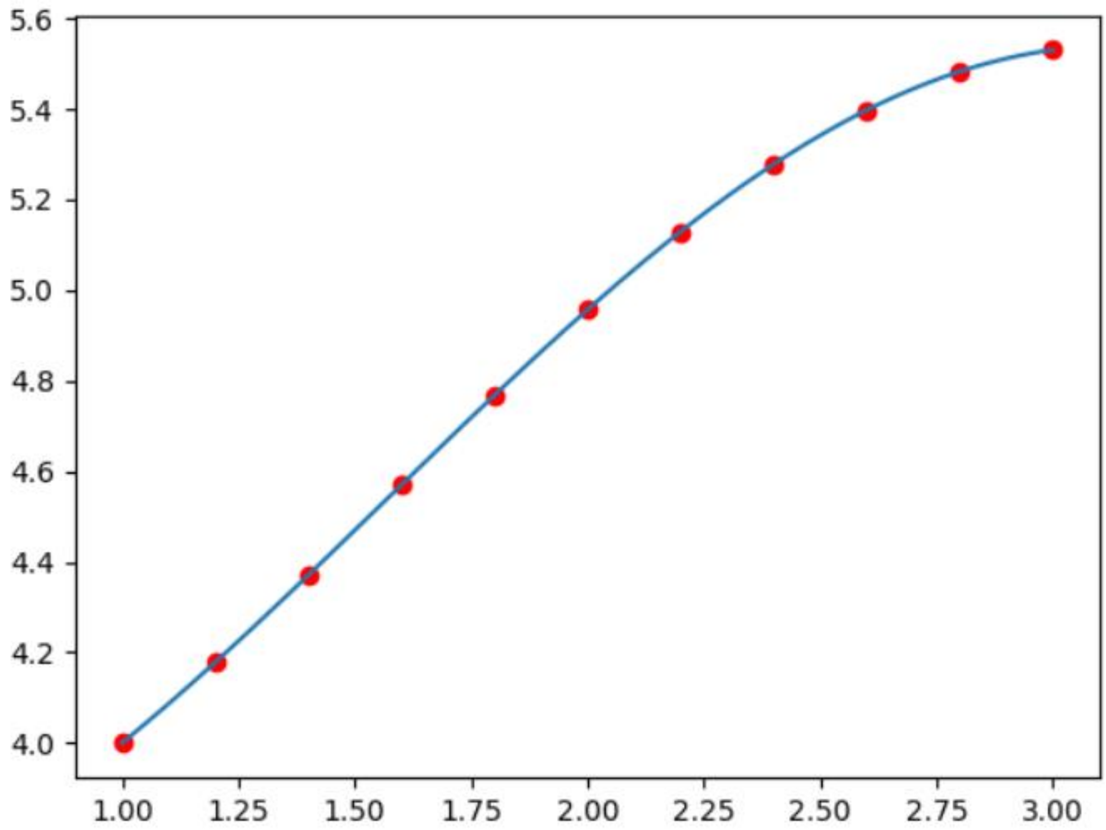
Точная функция: $y(x) = \frac{3}{4}e^{-2x} + \frac{1}{2}x^2 - \frac{1}{2}x + \frac{1}{4}$



Тест 3

```
Choose one of differential equations:
1) y' - x^2 - 2y = 0
2) y' =  sin(x)
3) y' = 1 + e^(-x) * cos(pi * x) - y
2
Input start points
x = 1
y = 4
Input end point
3
Input precision
5
```

Точная функция: $y(x) = 4 - \cos(x) + \cos(1)$



Вывод.

Дифференциальное уравнение (дальше ДУ) – уравнение, в которое входят производные функции и также могут входить сама функция, независимая переменная x и параметры. Найти решение ДУ – значит найти множество всех n раз дифференцируемых функций, которые удовлетворяют заданному уравнению. Обычно такое множество имеет вид $y =$

$f(x, C)$, где С – произвольная постоянная. Также такое множество называют общим решением ДУ.

Задача Коши – задача отыскания решения ДУ, удовлетворяющего заданному начальному условию $y(x_0) = y_0$. Решить задачу Коши означает найти общее решение ДУ, а потом подставить в него начальные условия и найти произвольные постоянные.

Методы приближенных вычислений такие, как представленные в вариантах к данной лабораторной работе, используются для решения задачи Коши, где ДУ имеет вид $y' = f(x, y)$. Зачастую на практике переменные в $y' = f(x, y)$ разделить невозможно, а если и возможно, то взять интеграл не получается, но при помощи данных методов приближенных вычислений можно с высокой точностью на некотором промежутке найти функцию $y = f(x)$, которая будет являться решением ДУ.

Данные методы делят на два типа:

1. Одношаговые (метод Эйлера и его модернизация, и метод Рунге-Кутта 4-го порядка)

2. Многошаговые (метод Адамса и метод Милна)

Различие между одношаговыми и многошаговыми методами состоит в том, что первые не используют ранее найденные значения, кроме тех, которые были найдены на предыдущем шаге, в то время как вторые для повышения точности ими пользуются. Однако здесь есть пара нюансов. Первый из них это то, что многошаговым методам для начала вычисления нужно иметь k первых значений (k – точность многошагового метода). Для их нахождения придется воспользоваться одношаговым методом (обычно им является метод Рунге-Кутта, так как он обеспечивает точность четвертого порядка, в то время как метод Эйлера имеет только точность первого порядка, а его модернизация – точность второго). Здесь возникает второй нюанс, ведь, если мы решим аппроксимировать функцию одношаговым методом с тем же количеством шагов, что и для многошагового, то точность упадет до точности одношагового метода, чего бы нам не хотелось допускать. Поэтому здесь есть два выхода:

1. Насчитывать первые k значений с помощью одношагового метода такого же порядка точности, что и многошаговый метод

2. Использовать методы более низкого порядка, разбивая на более мелкие шаги, чтобы через какое-то количество шагов получить результат нужной нам точности

В итоге получается, что в первом случае будет найдено решение небольшой точности, а во втором время работы метода заметно увеличится, так как асимптотика того же метода Рунге-Кутта составляет $O(\frac{b-a}{\sqrt[4]{\varepsilon}})$ (где a и b – концы промежутка, на котором нужно найти решения), а если рассмотреть на этом же промежутке выполнение усовершенствованного метода Эйлера (асимптотика – $O(\frac{b-a}{\sqrt{\varepsilon}})$), то окажется, что ему придется сделать количество итераций равное квадрату количества итераций метода Рунге-Кутта.