Федеральное государственное автономное образовательное учреждение высшего образования
«Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерной техники

Дисциплина: «Вычислительная математика»

Лабораторная работа №5
Вариант: Усовершенствованный метод Эйлера

Выполнил: Кизилов Степан Александрович,
группа P32312
Преподаватель: Перл Ольга Вячеславовна

Санкт-Петербург, 2023 год

1 Описание метода

Идея (и отличие от обычного метода Эйлера) состоит в том, что мы строим отрезок ломаной не по левому краю отрезка (касательной в этой точке), а по его центру, что улучшает приближение.
Тогда средняя точка:

$$x_z = x_i + h/2$$

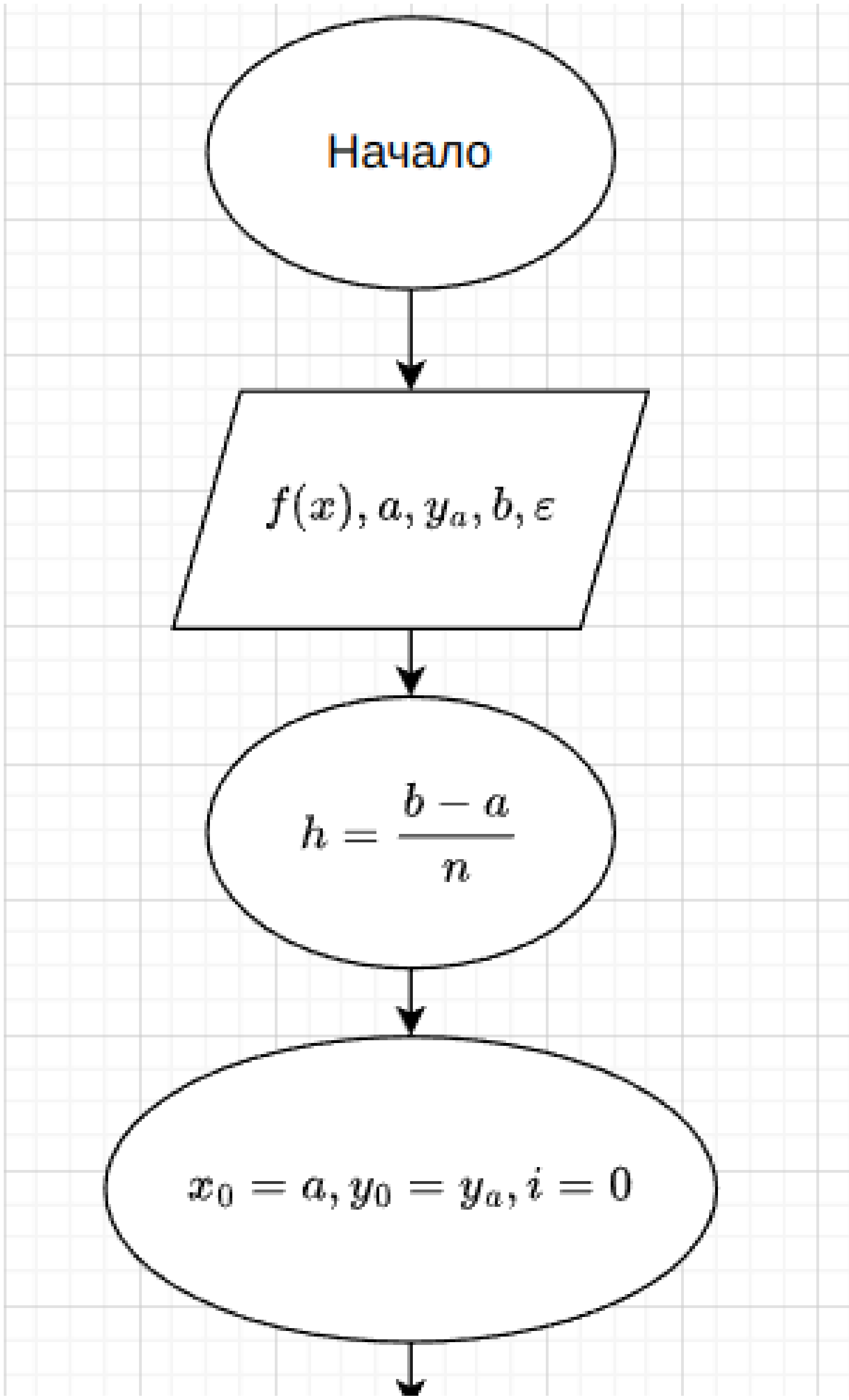
$$y_z = y_i + h/2 * f(x_i, y_i)$$

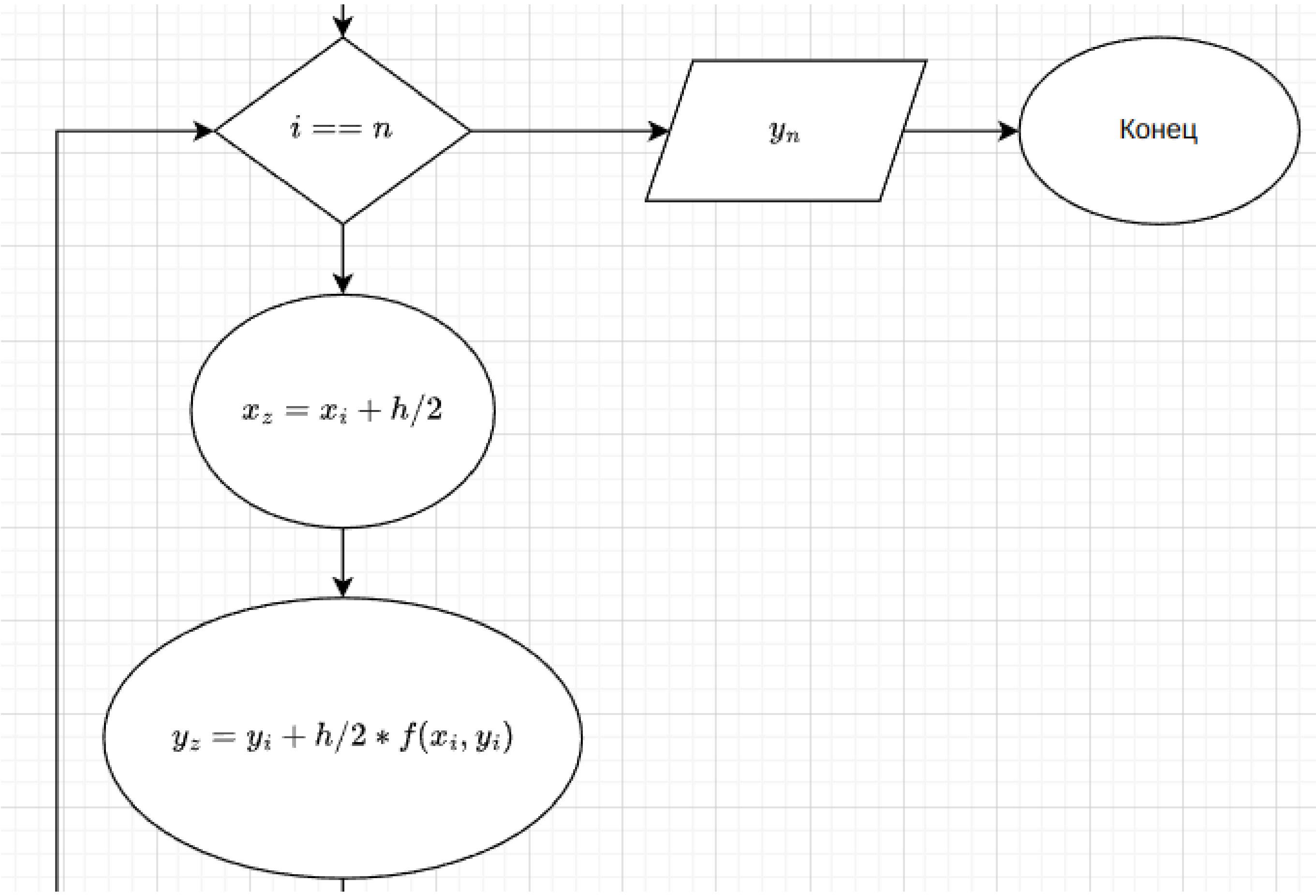
$$f_z = f(x_z, y_z)$$

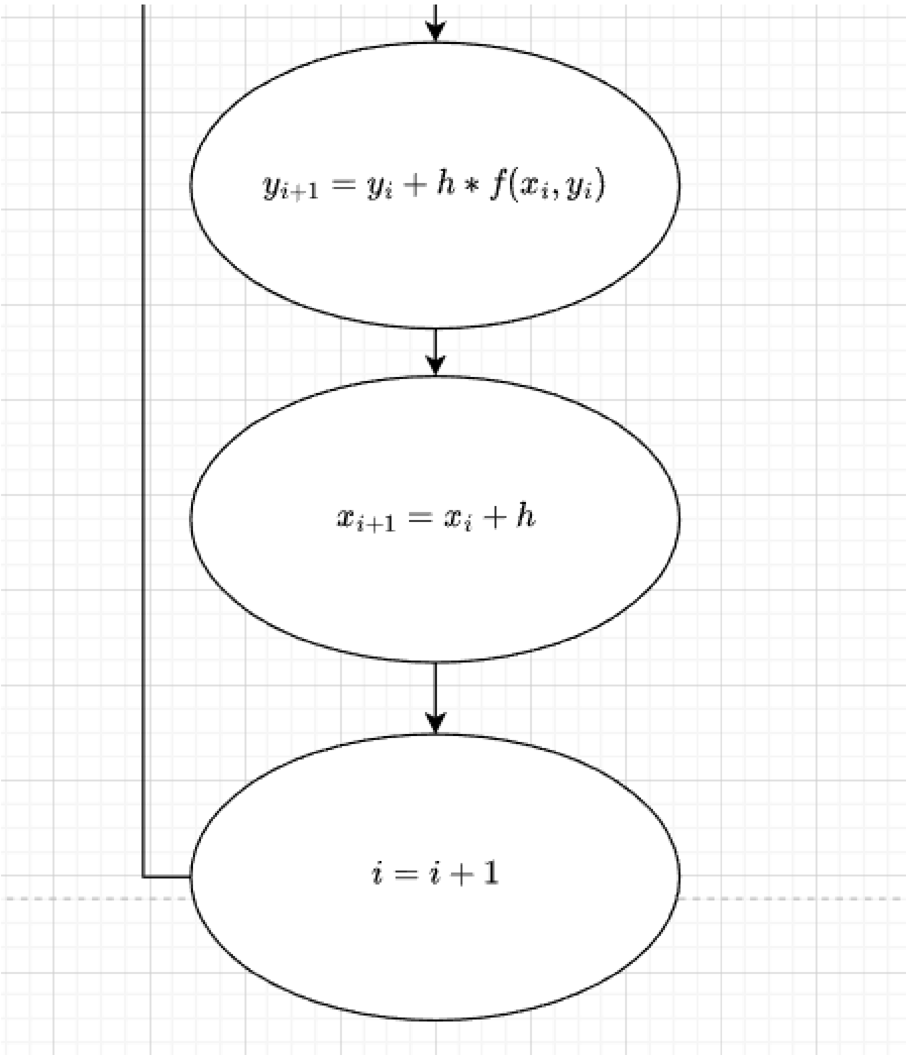
И затем уже делаем само приближение:

$$y_{i+1} = y_i + h * f_z$$

2 Блок-схема метода




$i == n$
y_n
Конец
$x_z = x_i + h/2$
$y_z = y_i + h/2 * f(x_i, y_i)$

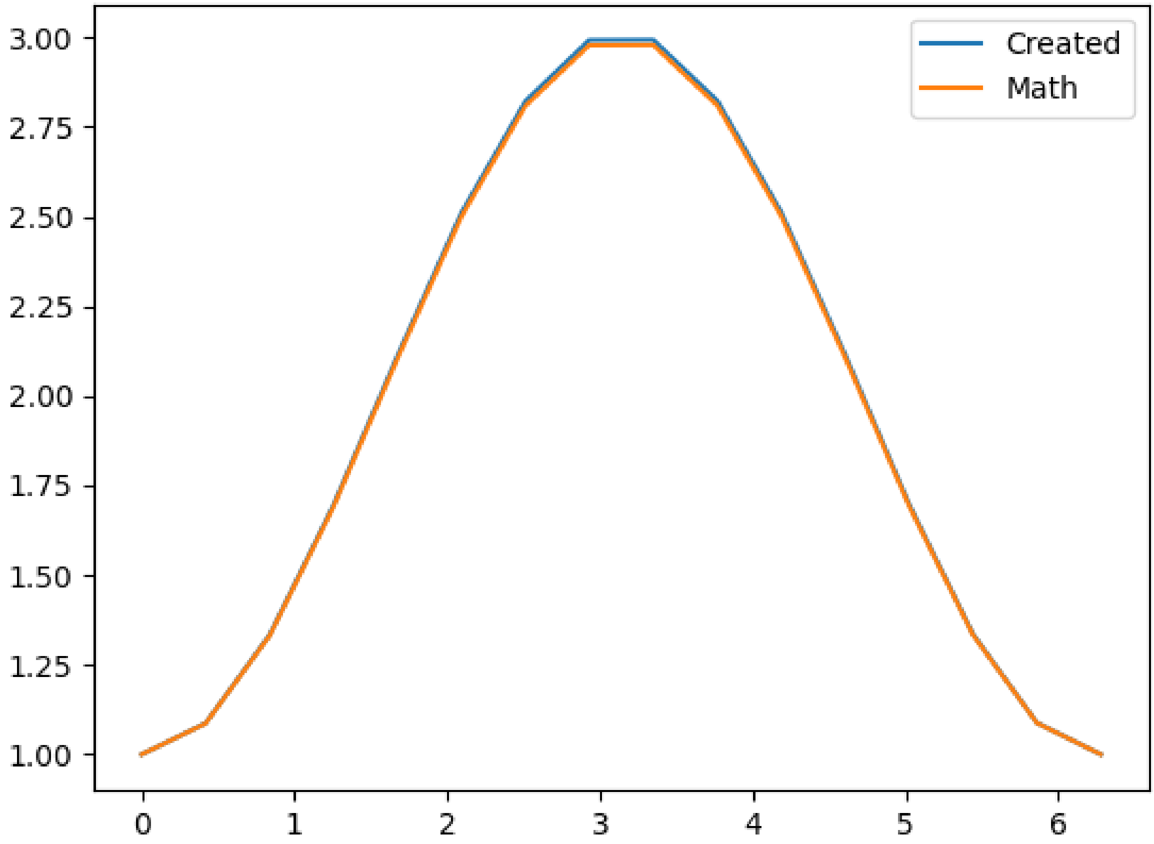

$y_{i+1} = y_i + h * f(x_i, y_i)$
$x_{i+1} = x_i + h$
$i = i + 1$

3 Исходный код

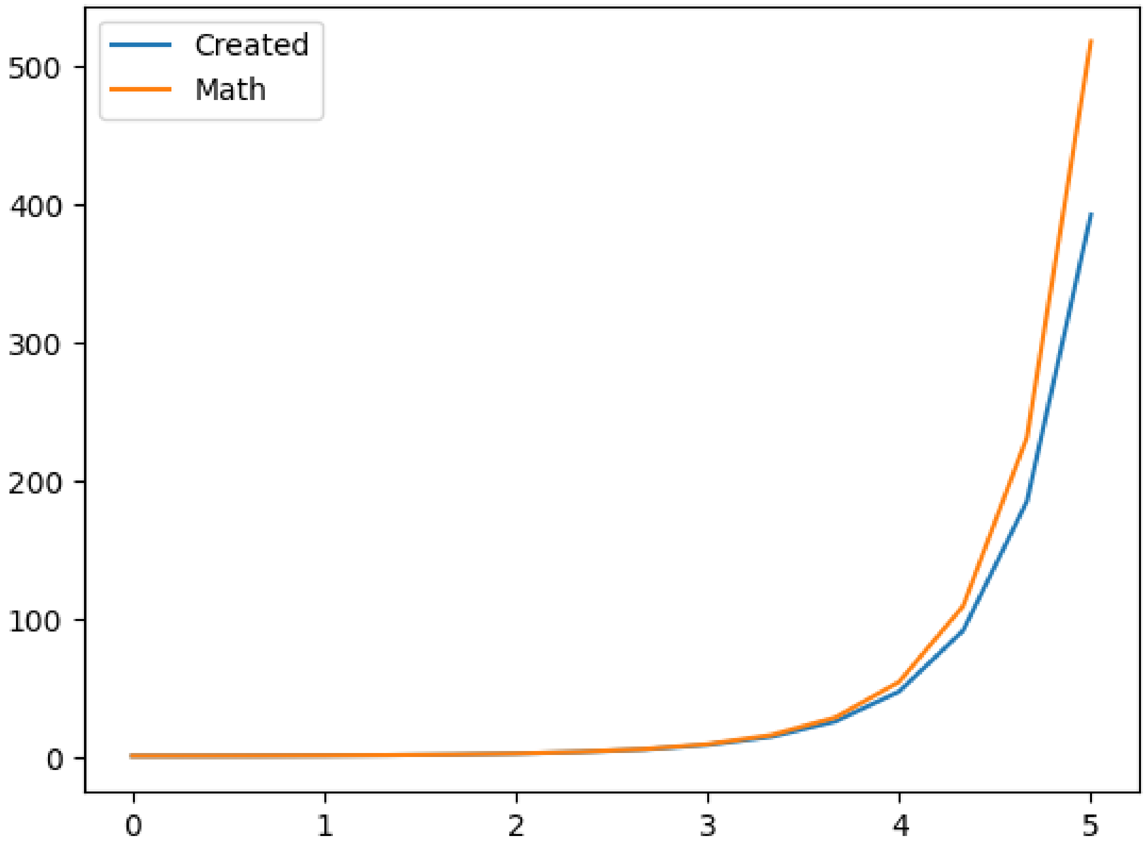
```
def solveByEulerImproved(loc_f, loc_epsilon, loc_a, loc_y_a, loc_b):
    func = Result.get_function(loc_f)

    steps = 15
    x = [loc_a]
    y = [loc_y_a]

    for _ in range(steps):
        h = (loc_b - loc_a) / steps

        x_z = x[-1] + h / 2
        y_z = y[-1] + h / 2 * (func(x[-1], y[-1]))

        y.append(y[-1] + h * func(x_z, y_z))
        x.append(x[-1] + h)

    import matplotlib.pyplot as plt

    plt.plot(x, y, label="Created")

    match loc_f:
        case 1:
            plt.plot(x, [-math.cos(i) + 2 for i in x], label="Math")
        case 2:
            plt.plot(x, [math.e ** (i * i / 4) for i in x], label="Math")

    plt.legend()
    plt.show()

    return y[-1]
```

4 Примеры и результаты работы

4.1 Пример 1



4.2 Пример 2



5 Вывод

Данный метод точнее чем обычный метод Эйлера за счёт лучшего приближения, но всё равно имеет недостаток в виде того, что ошибка продолжает постепенно накапливаться. Поэтому, если нужна достаточно хорошая точность, то можно, например, использовать метод Рунге-Кутты, который использует 4 приближения вместо 2.