Министерство образования Республики Беларусь
Учреждение образования «Белорусский государственный университет информатики и радиоэлектроники»

Факультет компьютерных систем и сетей
Кафедра информатики
Дисциплина «Методы численного анализа»

**ОТЧЕТ**

к лабораторной работе №9

на тему:

**«МЕТОДЫ ЭЙЛЕРА И РУНГЕ-КУТТА»**

БГУИР 1-40 04 01

Выполнил студент группы 253505
БЕКАРЕВ Станислав Сергеевич

______________________________
(дата, подпись студента)

Проверил доцент кафедры информатики
АНИСИМОВ Владимир Яковлевич

______________________________
(дата, подпись преподавателя)

Минск 2023

## Содержание

## Цель работы

- изучить решение задачи Коши для обыкновенных дифференциальных уравнений методом Эйлера и методом Рунге-Кутта;
- сравнить методы по трудоемкости, точности;
- составить программу реализации методов;
- выполнить тестовые примеры и проверить правильность работы программы

**ЗАДАНИЕ.** С помощью метода Эйлера, а затем метода Рунге-Кутта найти с точностью до 0.001 решения следующих уравнений на отрезке [0; 1].

$$y' = \frac{a\left(1-y^2\right)}{(1+m)x^2+y^2+1}, \quad y(0)=0,$$

где значения параметров $a$ и $m$ принимают следующие значения для вариантов $k$.

| k | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $m$ | 1.0 | 1.5 | 2.0 | 1.0 | 1.5 | 2.0 | 1.0 | 1.5 | 2.0 | 1.0 | 1.5 | 2.0 | 1.0 | 2.0 |
| $a$ | 0.5 | 0.7 | 0.9 | 1.1 | 1.3 | 0.5 | 0.7 | 0.9 | 1.1 | 1.3 | 0.5 | 0.7 | 0.9 | 1.0 |

**Вариант 3**

*Исходные данные*

```
Изначальное дифференциальное уравнение:
y' = (0.7 - 0.7*y**2)/(2.5*x**2 + y**2 + 1)
Изначальный шаг: 1.0
Заданая точночть: 0.001
Точка для проверки значений: X0 = 0.5
```

Примечание: x ** n (возвести x в степень n)

## Программная реализация

Код метода Эйлера:

```
def euler_method(func, f_y0, left, right, accuracy, epsilon):
    tempY = -666.
    while True:
        X, Y = [], []
        n = int((right - left) / accuracy)
        X.append(left)
        Y.append(f_y0)
        for i in range(n):
            X.append(X[i] + accuracy)
            Y.append(Y[i] + accuracy * func.subs([(x, X[i]), (y,
Y[i])]))
        if abs(tempY - Y[-1]) < epsilon:
            break
        tempY = Y[-1]
        accuracy /= 2
    return X, Y, accuracy
```

Код модифицированного метода Эйлера:

```
def modify_euler_method(func, f_y0, left, right, accuracy,
epsilon):
    tempY = -666.
    while True:
        X, Y = [], []
        n = int((right - left) / accuracy)
        X.append(left)
        Y.append(f_y0)
        for i in range(n):
            X.append(X[i] + accuracy)
            Y.append(Y[i] + accuracy * func.subs([(x, X[i] + 0.5
* accuracy),
                                                   (y, Y[i] + 0.5 * accuracy
* func.subs([(x, X[i]), (y, Y[i])]))]))
        if abs(tempY - Y[-1]) < epsilon:
            break
        tempY = Y[-1]
        accuracy /= 2
    return X, Y, accuracy
```

Код метода Рунге-Кутта:

```
def runge_kutta_method(func, f_y0, left, right, accuracy,
epsilon):
    tempY = -666.
    while True:
        X, Y = [], []
        K = [0., 0., 0., 0.]
        n = int((right - left) / accuracy)
        X.append(left)
        Y.append(f_y0)
        for i in range(n):
            K[0] = accuracy * func.subs([(x, X[i]), (y, Y[i])])
            K[1] = accuracy * func.subs([(x, X[i] + 0.5 *
accuracy), (y, Y[i] + 0.5 * K[0])])
            K[2] = accuracy * func.subs([(x, X[i] + 0.5 *
accuracy), (y, Y[i] + 0.5 * K[1])])
            K[3] = accuracy * func.subs([(x, X[i] + accuracy),
(y, Y[i] + K[2])])
            X.append(X[i] + accuracy)
            Y.append(Y[i] + (K[0] + 2*K[1]+ 2*K[2]+ K[3])/ 6)
        if abs(tempY - Y[-1]) < epsilon:
            break
        tempY = Y[-1]
        accuracy /= 2
    return X, Y, accuracy
```

## Полученные результаты

```
Изначальное дифференциальное уравнение:
y' = (0.9 - 0.9*y**2)/(3.0*x**2 + y**2 + 1)
Изначальный шаг: 1.0
Заданая точночть: 0.001
Точка для проверки значений: X0 = 0.5
Метод Эйлера
Метод достиг указанной точности при h = 0.001953125
Значение в точке X0:
0.34562148
Модернизированный метод Эйлера (h = 1.0)
Метод достиг указанной точности при h = 0.0625
Значение в точке X0:
0.34521608
Метод Рунге-Кутта (h = 1.0)
Метод достиг указанной точности при h = 0.25
Значение в точке X0:
0.34517993
```

*Тестовый пример 1*

```
Изначальное дифференциальное уравнение:
y' = x*y
y = exp(x**2/2)
Изначальный шаг: 1.0
Заданая точночть: 0.0001
Точка для проверки значений: X0 = 0.5
Значение функции в точке: 1.13314845306683
Метод Эйлера
Метод достиг указанной точности при h = 6.103515625e-05
Значение в точке X0:
1.13312972
Модернизированный метод Эйлера (h = 1.0)
Метод достиг указанной точности при h = 0.0078125
Значение в точке X0:
1.13314612
Метод Рунге-Кутта (h = 1.0)
Метод достиг указанной точности при h = 0.125
Значение в точке X0:
1.13314843
```

*Тестовый пример 2*

```
Изначальное дифференциальное уравнение:
y' = -tan(x)
y = log(cos(x))
Изначальный шаг: 1.0
Заданая точночть: 0.0001
Точка для проверки значений: X0 = 0.5
Значение функции в точке: -0.130584240443723
Метод Эйлера
Метод достиг указанной точности при h = 0.0001220703125
Значение в точке X0:
-0.13055090
Модернизированный метод Эйлера (h = 1.0)
Метод достиг указанной точности при h = 0.015625
Значение в точке X0:
-0.13058120
Метод Рунге-Кутта (h = 1.0)
Метод достиг указанной точности при h = 0.125
Значение в точке X0:
-0.13058449
```

## Вывод

В ходе выполнения лабораторной работы я изучил решение задачи Коши для обыкновенных дифференциальных уравнений методом Эйлера и методом Рунге-Кутта, написал программу их реализации на языке Python, правильность работы программы проверил на тестовых примерах.

На основании тестов можно сделать следующие выводы:

- Недостатком метода Эйлера является его не достаточно высокая точность. Достоинством метода Эйлера является его простота.
- Метод Рунге-Кутта 4-го порядка отличается очень высокой точностью. К определенным его недостаткам относится большая сложность и трудоемкость (на каждом шаге необходимо четырежды вычислять значения функции / вместо одного раза в методе Эйлера).