Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра информатики

Дисциплина: Методы численного анализа

**ОТЧЁТ**

к лабораторной работе

на тему

Методы Эйлера и Рунге-Кутта

Выполнил: студент группы 153503

Кончик Денис Сергеевич

Проверил: Анисимов Владимир Яковлевич

Минск 2022

**Содержание**

[1. ЦЕЛЬ РАБОТЫ 3](#_Toc120059950)

[2. ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ 3](#_Toc120059951)

[3. АЛГОРИТМ РЕШЕНИЯ 8](#_Toc120059952)

[4. ПРОГРАММНАЯ РЕАЛИЗАЦИЯ 9](#_Toc120059953)

[5. ТЕСТОВЫЕ ПРИМЕРЫ 12](#_Toc120059954)

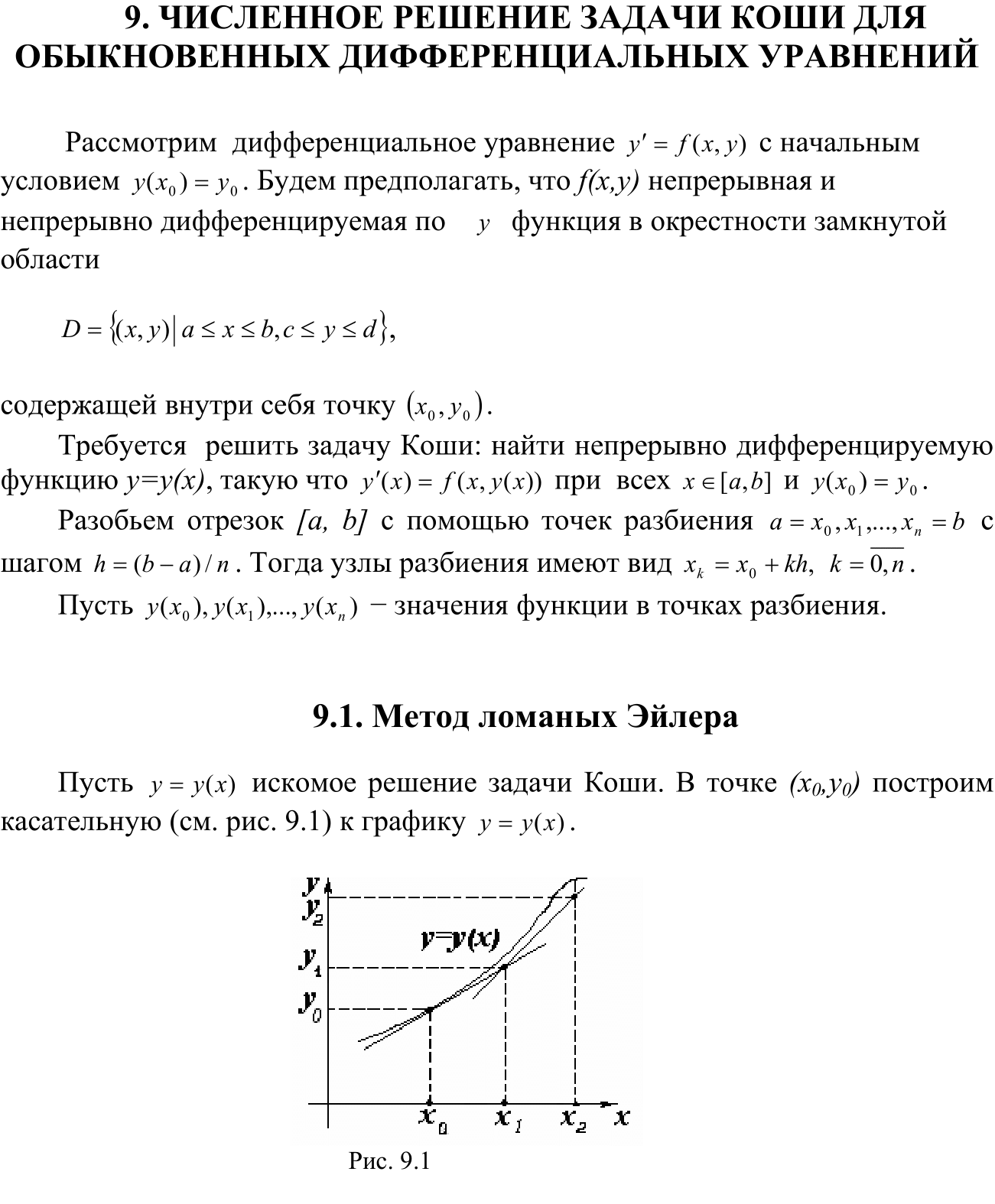
[6. ЗАДАНИЕ 14](#_Toc120059955)

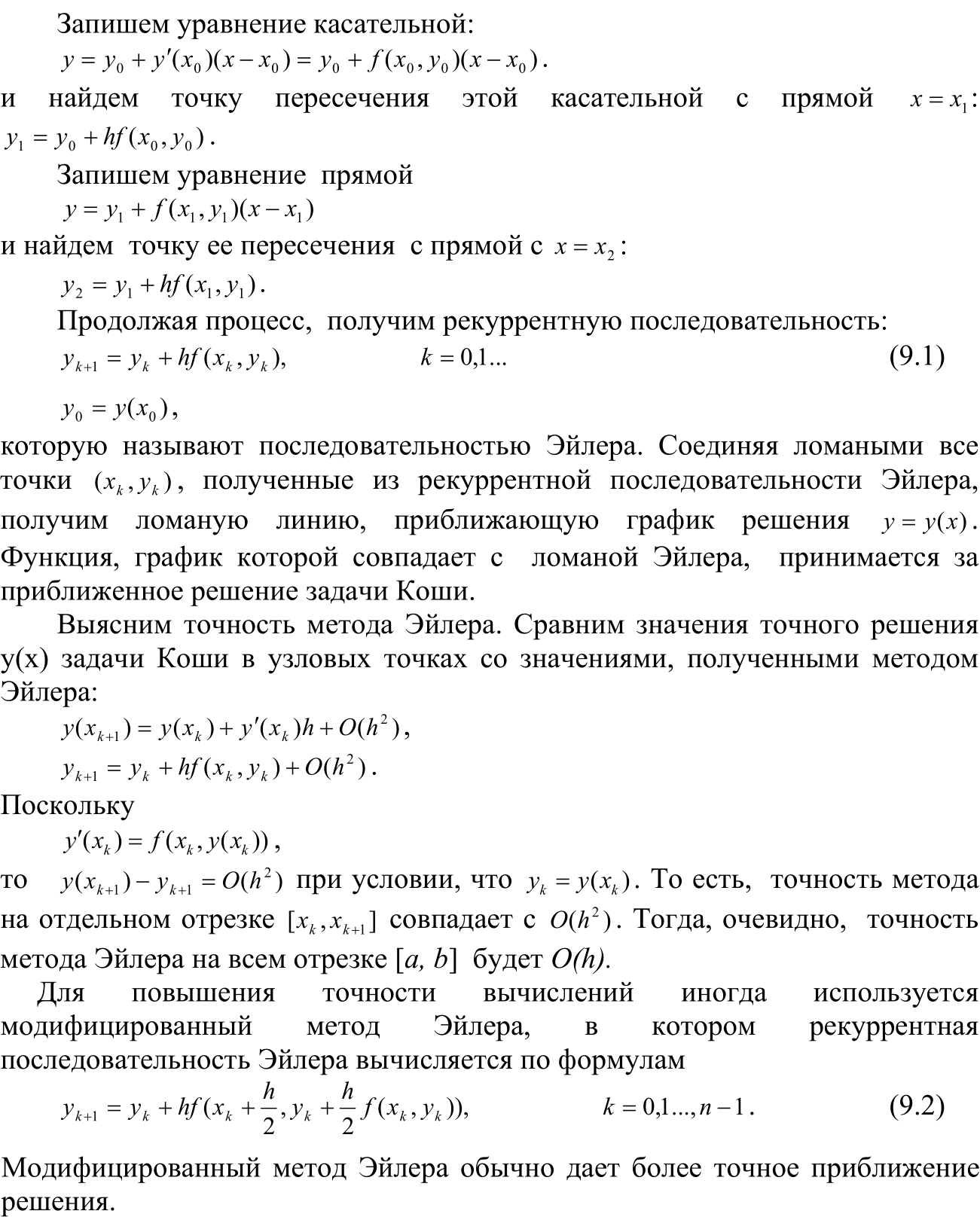
[7. ВЫВОД 19](#_Toc120059956)

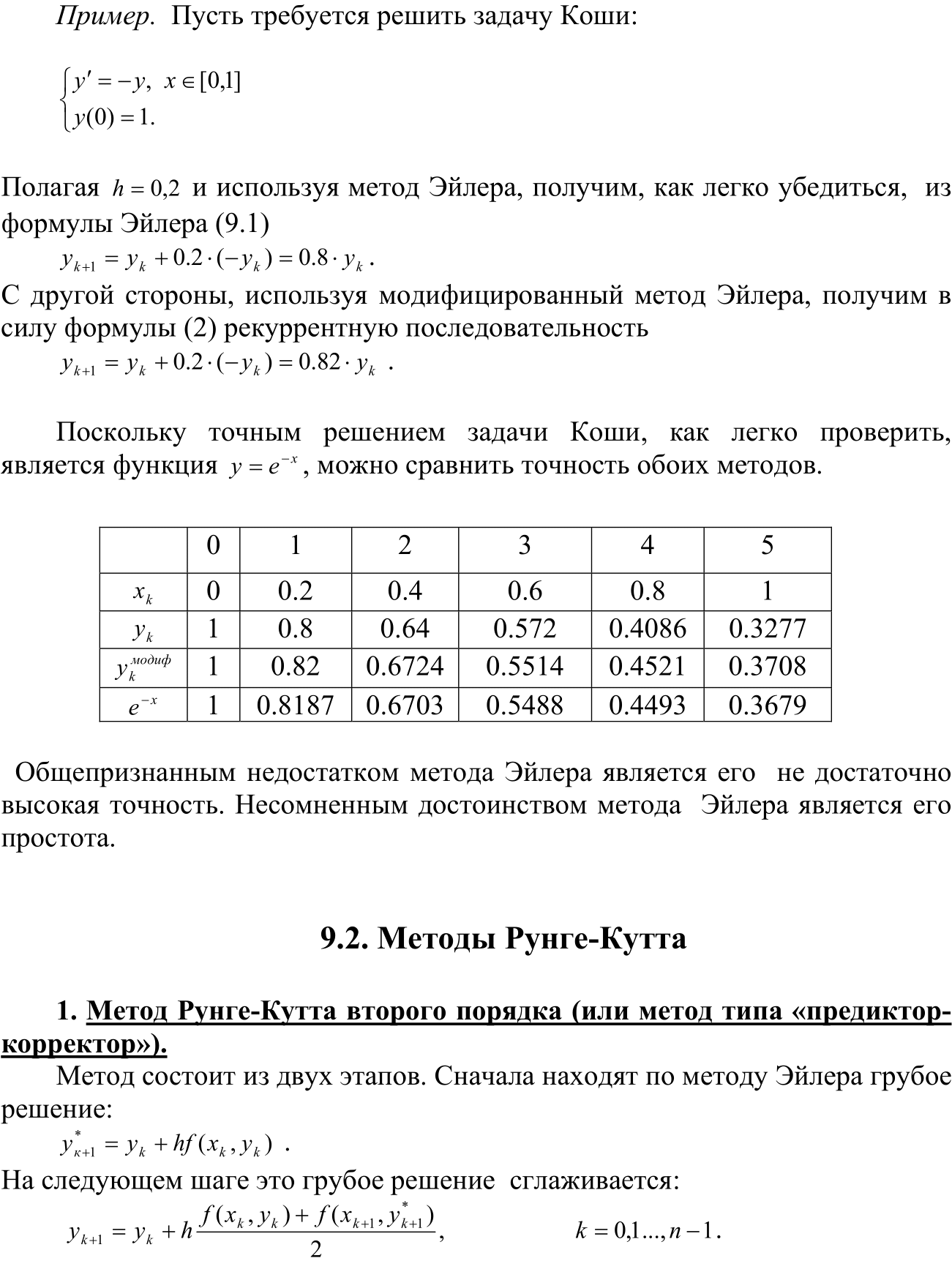
1. **ЦЕЛЬ РАБОТЫ**

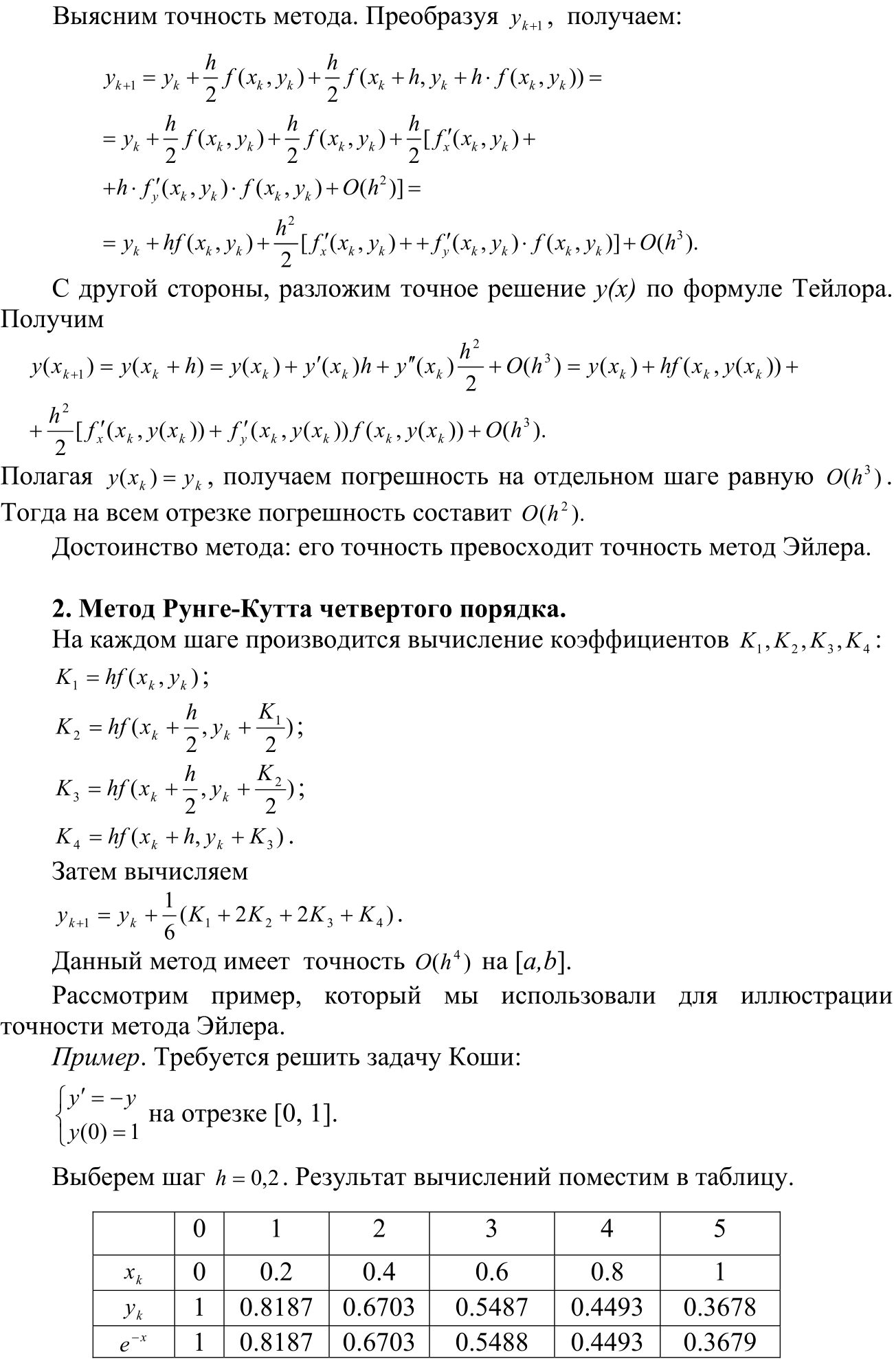
Изучить решение задачи Коши для обыкновенных дифференциальных уравнений методом Эйлера и методом Рунге-Кутта.

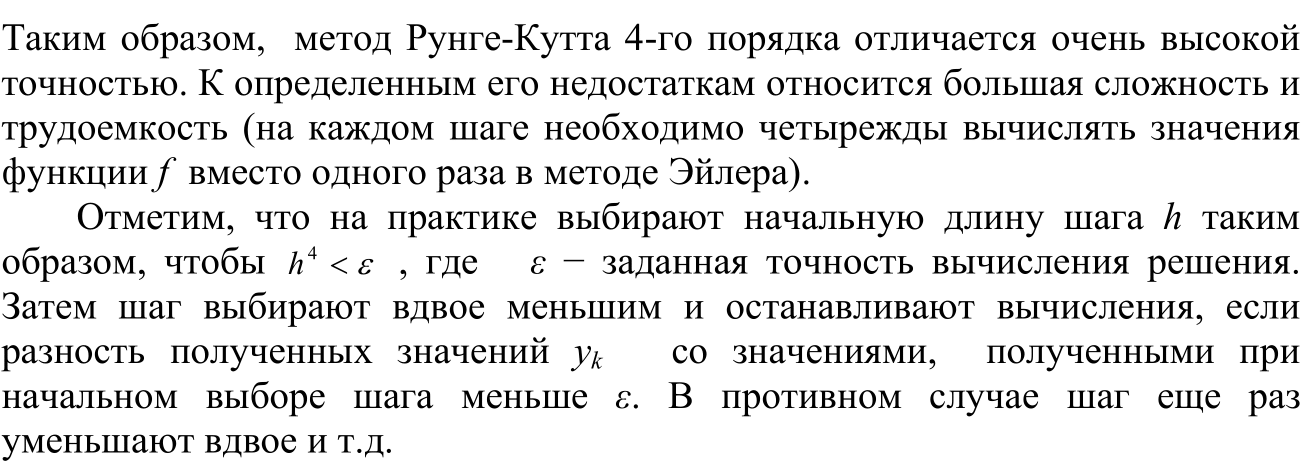
1. **ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ**











# **АЛГОРИТМ РЕШЕНИЯ**



# **ПРОГРАММНАЯ РЕАЛИЗАЦИЯ**

import numpy as np

import matplotlib.pyplot as plt

import sympy

plot\_dots = 10\*\*3

eps = 10\*\*-3

# Задание

m, a = 1.5, 0.5

x0, y0 = 0, 0

L, R = 0, 1

def f(x, y): # y' = f(x,y)

return (a \* (1 - y\*\*2)) / ((1 + m) \* x\*\*2 + y\*\*2 + 1)

def fderx(x, y):

numerator = 2\*a\*x+2\*a\*m\*x-2\*a\*x\*(y\*\*2)-2\*a\*m\*x\*(y\*\*2)

denominator = ((x\*\*2)+m\*(x\*\*2)+(y\*\*2)+1)\*\*2

return -numerator/denominator

def fdery(x, y):

numerator = -2\*a\*(x\*\*2)\*y-2\*a\*m\*(x\*\*2)\*y-4\*a\*y

denominator = ((x\*\*2)+m\*(x\*\*2)+(y\*\*2)+1)\*\*2

return numerator / denominator

# Метод Эйлера

def EulerMethod(x, n):

y\_list = [y0]

h = (x - x0) / n

for k in range(n):

x\_k = x0 + k \* h

y\_k = y\_list[-1]

y\_list.append(y\_k + h \* f(x\_k, y\_k))

return y\_list

# Модифицированный метод Эйлера

def EulerModifiedMethod(x, n):

y\_list = [y0]

h = (x - x0) / n

for k in range(n):

x\_k = x0 + k \* h

y\_k = y\_list[-1]

y\_list.append(y\_k + h \* f(x\_k + (h / 2), y\_k + (h / 2) \* f(x\_k, y\_k)))

return y\_list

# Метод Рунге-Кутта 2 порядка

def RungeKuttaMethod2(x, n):

y\_list = [y0]

h = (x - x0) / n

for k in range(n):

x\_k = x0 + k \* h

y\_k = y\_list[-1]

y\_list.append(y\_k + h \* f(x\_k, y\_k) + ((h\*\*2) / 2) \* (fderx(x\_k, y\_k) + fdery(x\_k, y\_k) \* f(x\_k, y\_k)))

return y\_list

# Метод Рунге-Кутта 4 порядка

def RungeKuttaMethod4(x, n):

y\_list = [y0]

h = (x - x0) / n

for k in range(n):

x\_k = x0 + k \* h

y\_k = y\_list[-1]

K1 = h \* f(x\_k, y\_k)

K2 = h \* f(x\_k + h / 2, y\_k + K1 / 2)

K3 = h \* f(x\_k + h / 2, y\_k + K2 / 2)

K4 = h \* f(x\_k + h, y\_k + K3)

y\_list.append(y\_k + 1/6 \* (K1 + 2 \* K2 + 2 \* K3 + K4))

return y\_list

# Вычислить Y по методу и X

def CalculateY(method, x):

n = 1 # Количество узлов от x0 до x для точности eps

while True:

y\_list = method(x, n)

y\_list\_correctly = method(x, n \* 2)

max\_delta = max(abs(y\_list\_correctly[2 \* i] - y\_list[i]) for i in range(n + 1))

if (max\_delta < eps):

return y\_list\_correctly[-1], n \* 2

else:

n \*= 2

# Создать список Y по методу и списку X

def CalculateListY(method, x\_list):

y\_list = [] # Список игреков

n\_list = [] # Список n (числа узлов) для каждой точки

for x in x\_list:

y, n = CalculateY(method, x)

y\_list.append(y)

n\_list.append(n)

return y\_list, max(n\_list), sum(n\_list) / len(x\_list)

print(f"Количество точек для построения графика: {plot\_dots}")

print(f"Точность: {eps}")

#print(f"Функция: {func}")

x\_list = []

for i in range(plot\_dots + 1):

x\_list.append(L + (R - L) / plot\_dots \* i)

print("\nN\_max – максимальное количество узлов для одной из точек, \nнеобходимое для достижения заданной точности")

print("N\_middle – среднее количество узлов по всем точкам при \nдостижении заданной точности")

y\_list, N\_max, N\_middle = CalculateListY(EulerMethod, x\_list)

print("\nМетод Эйлера O(h):")

print(f"N\_max: {N\_max}")

print(f"N\_middle: {int(N\_middle)}")

print("На графике: красный")

plt.plot(x\_list, y\_list, 'red', linewidth = 4)

y\_list, N\_max, N\_middle = CalculateListY(EulerModifiedMethod, x\_list)

print("\nМетод Эйлера (модифицированный) O(h^2):")

print(f"N\_max: {N\_max}")

print(f"N\_middle: {int(N\_middle)}")

print("На графике: голубой")

plt.plot(x\_list, y\_list, '--b', linewidth = 3)

y\_list, N\_max, N\_middle = CalculateListY(RungeKuttaMethod2, x\_list)

print("\nМетод Рунге-Кутта 2 порядка O(h^2):")

print(f"N\_max: {N\_max}")

print(f"N\_middle: {int(N\_middle)}")

print("На графике: желтый")

plt.plot(x\_list, y\_list, '-.y', linewidth = 3)

y\_list, N\_max, N\_middle = CalculateListY(RungeKuttaMethod4, x\_list)

print("\nМетод Рунге-Кутта 4 порядка O(h^4):")

print(f"N\_max: {N\_max}")

print(f"N\_middle: {int(N\_middle)}")

print("На графике: черный")

plt.plot(x\_list, y\_list, ':k', linewidth = 3)

plt.show()

**5. ТЕСТОВЫЕ ПРИМЕРЫ**

**Тестовый пример 1.1.**

С помощью метода Эйлера*1*, модифицированного метода Эйлера*2*, метода Рунге-Кутта 2-го порядка*3*, Рунге-Кутта 4-го порядка*4* найти с заданной точностью решение заданного уравнения на заданном отрезке.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ДУ | Начальное условие | | | Отрезок | Решение | | Точность | | Точек разбиения |
|  |  | | |  |  | |  | |  |
|  | | | | | | | | | |
| 1 – красный, 2 – синий, 3 – желтый, 4 – черный | | | | | | | | | |
| Количество точек разбиения в методе | | | | | | | | | |
|  | | Эйлер | мод. Эйлер | | | Рунге-Кутт (2) | | Рунге-Кутт (4) | |
| Максимальное | | 512 | 32 | | | 32 | | 8 | |
| Среднее | | 267 | 19 | | | 19 | | 4 | |

**Тестовый пример 1.2.**

Для условия предыдущего задания найти значения решения задачи Коши в заданных точках.

|  |  |  |  |
| --- | --- | --- | --- |
| Метод Эйлера O(h) | | | |
| x | 0.3 | 0.9 | 1.4 |
|  |  |  |  |
| Метод(х) |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Модифицированный метод Эйлера O(h^2) | | | |
| x | 0.3 | 0.9 | 1.4 |
|  |  |  |  |
| Метод(х) |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Метод Рунге-Кутта 2 порядкаO(h^2) | | | |
| x | 0.3 | 0.9 | 1.4 |
|  |  |  |  |
| Метод (х) |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Метод Рунге-Кутта 4 порядкаO(h^4) | | | |
| x | 0.3 | 0.9 | 1.4 |
|  |  |  |  |
| Метод (х) |  |  |  |
|  |  |  |  |

**Тестовый пример 2.1.**

С помощью метода Эйлера*1*, модифицированного метода Эйлера*2*, метода Рунге-Кутта 2-го порядка*3*, Рунге-Кутта 4-го порядка*4* найти с заданной точностью решение заданного уравнения на заданном отрезке.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ДУ | Начальное условие | | | Отрезок | Решение | | Точность | | Точек разбиения |
|  |  | | |  |  | |  | |  |
|  | | | | | | | | | |
| 1 – красный, 2 – синий, 3 – желтый, 4 – черный | | | | | | | | | |
| Количество точек разбиения в методе | | | | | | | | | |
|  | | Эйлер | мод. Эйлер | | | Рунге-Кутт (2) | | Рунге-Кутт (4) | |
| Максимальное | | 8192 | 128 | | | 256 | | 2 | |
| Среднее | | 1253 | 39 | | | 77 | | 2 | |

**Тестовый пример 2.2.**

Для условия предыдущего задания найти значения решения задачи Коши в заданных точках.

|  |  |  |  |
| --- | --- | --- | --- |
| Метод Эйлера O(h) | | | |
| x | 0 | 0.7 | 1.7 |
|  |  |  |  |
| Метод(х) |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Модифицированный метод Эйлера O(h^2) | | | |
| x | 0 | 0.9 | 1.4 |
|  |  |  |  |
| Метод(х) |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Метод Рунге-Кутта 2 порядкаO(h^2) | | | |
| x | 0 | 0.9 | 1.4 |
|  |  |  |  |
| Метод (х) |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Метод Рунге-Кутта 4 порядкаO(h^4) | | | |
| x | 0 | 0.9 | 1.4 |
|  |  |  |  |
| Метод (х) |  |  |  |
|  |  |  |  |

**Тестовый пример 3.1.**

С помощью метода Эйлера*1*, модифицированного метода Эйлера*2*, метода Рунге-Кутта 2-го порядка*3*, Рунге-Кутта 4-го порядка*4* найти с заданной точностью решение заданного уравнения на заданном отрезке.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ДУ | Начальное условие | | Отрезок | Решение | | Точность | | Точек разбиения |
|  |  | |  |  | |  | |  |
|  | | | | | | | | |
| 1 – красный, 2 – синий, 3 – желтый, 4 – черный | | | | | | | | |
| Количество точек разбиения в методе | | | | | | | | |
|  | | Эйлер | мод. Эйлер | | Рунге-Кутт (2) | | Рунге-Кутт (4) | |
| Макс. | | 16384 | 256 | | 256 | | 16 | |
| Среднее | | 3063 | 54 | | 64 | | 5 | |

**Тестовый пример 3.2.**

Для условия предыдущего задания найти значения решения задачи Коши в заданных точках.

|  |  |  |  |
| --- | --- | --- | --- |
| Метод Эйлера O(h) | | | |
| x | -0.5 | 0.5 | 1.5 |
|  |  |  |  |
| Метод(х) |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Модифицированный метод Эйлера O(h^2) | | | |
| x | -0.5 | 0.5 | 1.4 |
|  |  |  |  |
| Метод(х) |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Метод Рунге-Кутта 2 порядкаO(h^2) | | | |
| x | -0.5 | 0.5 | 1.4 |
|  |  |  |  |
| Метод (х) |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Метод Рунге-Кутта 4 порядкаO(h^4) | | | |
| x | -0.5 | 0.5 | 1.4 |
|  |  |  |  |
| Метод (х) |  |  |  |
|  |  |  |  |

6. ЗАДАНИЕ

Вариант 11

С помощью метода Эйлера1, модифицированного метода Эйлера2, методов Рунге-Кутта3, 4 (2 и 4 порядка) найти с точностью до 0.001 решение уравнения на отрезке [0; 1]. Сравнить результаты.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ДУ | Начальное условие | | | Отрезок | Точность | | Точек разбиения | |
|  |  | | |  |  | |  | |
|  | | | | | | | | |
| 1 – красный, 2 – синий, 3 – желтый, 4 – черный | | | | | | | | |
| Количество точек разбиения в методе | | | | | | | | |
|  | | Эйлер | мод. Эйлер | | | Рунге-Кутт (2) | | Рунге-Кутт (4) |
| Максимальное | | 256 | 16 | | | 32 | | 4 |
| Среднее | | 93 | 5 | | | 12 | | 2 |

# **7. ВЫВОД**

Таким образом, в ходе выполнения лабораторной работы был изучен метод Эйлера, модифицированный метод Эйлера, методы Рунге-Кутта (2 и 4 порядка) для решения задачи Коши ОДУ 1 порядка. Составлен алгоритм и программа, проверена правильность её работы на тестовых примерах, с заданной точностью построены графики решения задачи Коши.

Исходя из тестовых примеров и задания и учитывая количество необходимых точек разбиения отрезка для достижения заданной точности, можно сделать вывод о трудоемкости методов: метод Эйлера самый трудоемкий, для модифицированного метода Эйлера и метода Рунге-Кутта 2 порядка количества точек разбиения – числа одного порядка и меньше, чем у метода Эйлера. Самый эффективным методом оказался метод Рунге-Кутта 4 порядка.