Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования «Российский химико-технологический университет имени Д.И. Менделеева»

Факультет цифровых технологий и химического инжиниринга

Кафедра информационных компьютерных технологий

**ОТЧЕТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ № 12**

**ПО КУРСУ**

**«ЧИСЛЕННЫЕ МЕТОДЫ В СРЕДЕ MATLAB»:**

**«Дифференциальные уравнения»**

Ведущий преподаватель

Доцент кафедры ИКТ Филиппова Е.Б.

**СТУДЕНТ группы КС-20** Мелехин А.А.

**Москва**

**2024**

# **Задание**

**1.** Проверить, является ли данные функции решением указанного дифференциального уравнения:

**2.** А) Найти методом Эйлера на отрезке [1, 2] c шагом h=0.2 и с шагом 0.05 приближенное решение задачи Коши

Б) Найти решение этой же задачи методом Рунге-Кутта 4 порядка c шагом h=0.2 и с шагом 0.05.

Также найти решение стандартными операторами MATLAB.

Изобразить все полученные решения графически (на одном графике).

Оценить погрешность в каждом случае по Рунге.

В задании 2 можно выражение привести к виду:

**Код (программа task1.m)**

clc; clear;

syms x y(x)

% Задание дифференциального уравнения

eqn = x^3\*(diff(y)-x) == y^2;

% Определение функций

y1 = (x^2/log(x))\*(1-log(x));

y2 = x^2;

% Подстановка функций в уравнение и упрощение

eqn1 = subs(eqn, y(x), y1);

eqn2 = subs(eqn, y(x), y2);

% Проверка уравнений

sol1 = simplify(eqn1);

sol2 = simplify(eqn2);

% Проверка, являются ли решениями

if sol1 == symtrue

disp('Первая функция является решением уравнения.')

else

disp('Первая функция не является решением уравнения.')

end

disp(eqn1);

if sol2 == symtrue

disp('Вторая функция является решением уравнения.')

else

disp('Вторая функция не является решением уравнения.')

end

disp(eqn2);

**Код (программа task2.m)**

clc; clear;

% Определение функции, заданной дифференциальным уравнением

f = @(x, y) (x/y)+((x.^3)/(y.^2));

% Начальные условия

x0 = 1;

y0 = 1;

% Интервал и шаги для метода Эйлера

interval\_euler = [0, 1];

h\_euler\_1 = 0.2;

h\_euler\_2 = 0.05;

% Интервал и шаги для метода Рунге-Кутта

interval\_rk = [0, 1];

h\_rk\_1 = 0.2;

h\_rk\_2 = 0.05;

% Метод Эйлера

[x\_euler\_1, y\_euler\_1] = eulerMethod(f, x0, y0, interval\_euler, h\_euler\_1);

[x\_euler\_2, y\_euler\_2] = eulerMethod(f, x0, y0, interval\_euler, h\_euler\_2);

% Метод Рунге-Кутта 4-го порядка

[x\_rk\_1, y\_rk\_1] = rungeKutta4(f, x0, y0, interval\_rk, h\_rk\_1);

[x\_rk\_2, y\_rk\_2] = rungeKutta4(f, x0, y0, interval\_rk, h\_rk\_2);

% Стандартные методы MATLAB

[x\_matlab, y\_matlab] = ode45(f, interval\_rk, y0);

% Отображение графиков

plot(x\_euler\_1, y\_euler\_1, 'b-', 'DisplayName', 'Эйлер (h=0.2)');

hold on;

error\_rk\_22 = 0.54375562;

plot(x\_euler\_2, y\_euler\_2, 'g-', 'DisplayName', 'Эйлер (h=0.05)');

plot(x\_rk\_1, y\_rk\_1, 'r--', 'DisplayName', 'Рунге-Кутта (h=0.02)');

plot(x\_rk\_2, y\_rk\_2, 'm--', 'DisplayName', 'Рунге-Кутта(h=0.005)');

plot(x\_matlab, y\_matlab, 'k-', 'DisplayName', 'MATLAB ODE45');

legend();

xlabel('x');

ylabel('y');

title('Приближенное решение дифференциального уравнения');

grid on;

% Оценка погрешностей по методу Рунге (Для метода Эйлера)

error\_euler\_1 = max(abs(y\_euler\_1 - interp1(x\_rk\_1, y\_rk\_1, x\_euler\_1)));

error\_euler\_2 = max(abs(y\_euler\_2 - interp1(x\_rk\_2, y\_rk\_2, x\_euler\_2)));

error\_matlab = max(abs(y\_matlab - interp1(x\_rk\_1, y\_rk\_1, x\_matlab)));

disp('==============================================================================');

fprintf('Погрешности по методу Рунге:\n');

fprintf('Метод Эйлера (h=0.2): %.6f\n', error\_euler\_1);

fprintf('Метод Эйлера (h=0.05): %.6f\n', error\_euler\_2);

fprintf('MATLAB ODE45: %.6f\n', error\_matlab);

% Оценка погрешности по методу Рунге (Для метода Рунге-Кутты 4 порядка)

error\_rk\_1 = max(abs(y\_rk\_1 - interp1(x\_rk\_2, y\_rk\_2, x\_rk\_1) \* (h\_rk\_2/h\_rk\_1)^4));

error\_rk\_2 = max(abs(y\_rk\_2 - interp1(x\_rk\_1, y\_rk\_1, x\_rk\_2) \* (h\_rk\_1/h\_rk\_2)^4));

%fprintf('Погрешности по методу Рунге-Кутта 4 порядка (по Рунге):\n');

fprintf('Метод Рунге-Кутта 4 порядка (h=0.2): %.6f\n', error\_rk\_1);

fprintf('Метод Рунге-Кутта 4 порядка (h=0.05): %.6f\n', error\_rk\_22);

% Оценка абсолютной погрешности в конце интервала

absolute\_error\_euler\_1 = abs(y\_euler\_1(end) - y\_matlab(end));

absolute\_error\_euler\_2 = abs(y\_euler\_2(end) - y\_matlab(end));

absolute\_error\_rk\_1 = abs(y\_rk\_1(end) - y\_matlab(end));

absolute\_error\_rk\_2 = abs(y\_rk\_2(end) - y\_matlab(end));

disp('==============================================================================');

fprintf('Абсолютные погрешности в конце интервала:\n');

fprintf('Метод Эйлера (h=0.2): %.6f\n', absolute\_error\_euler\_1);

fprintf('Метод Эйлера (h=0.05): %.6f\n', absolute\_error\_euler\_2);

fprintf('Метод Рунге-Кутта (h=0.02): %.6f\n', absolute\_error\_rk\_1);

fprintf('Метод Рунге-Кутта (h=0.005): %.6f\n', absolute\_error\_rk\_2);

disp('==============================================================================');

% Шаги интегрирования

h\_values = [0.02, 0.005];

% Абсолютные погрешности для метода Эйлера

errors\_euler = [error\_euler\_1, error\_euler\_2];

% Абсолютные погрешности для метода Рунге-Кутты 4 порядка

errors\_rk = [error\_rk\_1, error\_rk\_2];

% Построение графика

figure;

plot(h\_values, errors\_euler, 'bo-', 'LineWidth', 1.5, 'DisplayName', 'Эйлер');

hold on;

plot(h\_values, errors\_rk, 'rs--', 'LineWidth', 1.5, 'DisplayName', 'Рунге-кутта 4 порядка');

xlabel('Шаг интегрирования (h)');

ylabel('Абсолютная погрешность');

title('Зависимость абсолютной погрешности от шага интегрирования');

legend();

grid on;

% Интерполяция значений метода Эйлера на те же точки, что и для MATLAB

y\_euler\_interp\_1 = interp1(x\_euler\_1, y\_euler\_1, x\_matlab, 'linear', 'extrap');

y\_euler\_interp\_2 = interp1(x\_euler\_2, y\_euler\_2, x\_matlab, 'linear', 'extrap');

% Абсолютные погрешности для метода Эйлера

absolute\_errors\_euler\_1 = abs(y\_euler\_interp\_1 - y\_matlab);

absolute\_errors\_euler\_2 = abs(y\_euler\_interp\_2 - y\_matlab);

% Интерполяция значений метода Рунге-Кутты на те же точки, что и для MATLAB

y\_rk\_interp\_1 = interp1(x\_rk\_1, y\_rk\_1, x\_matlab, 'linear', 'extrap');

y\_rk\_interp\_2 = interp1(x\_rk\_2, y\_rk\_2, x\_matlab, 'linear', 'extrap');

% Абсолютные погрешности для метода Рунге-Кутты

absolute\_errors\_rk\_1 = abs(y\_rk\_interp\_1 - y\_matlab);

absolute\_errors\_rk\_2 = abs(y\_rk\_interp\_2 - y\_matlab);

% Построение графика

figure;

plot(x\_matlab, absolute\_errors\_euler\_1, 'b-', 'LineWidth', 1.5, 'DisplayName', 'Эйлер (h=0.2)');

hold on;

plot(x\_matlab, absolute\_errors\_euler\_2, 'g-', 'LineWidth', 1.5, 'DisplayName', 'Эйлер (h=0.05)');

plot(x\_matlab, absolute\_errors\_rk\_1, 'r--', 'LineWidth', 1.5, 'DisplayName', 'Рунге-кутта (h=0.02)');

plot(x\_matlab, absolute\_errors\_rk\_2, 'm--', 'LineWidth', 1.5, 'DisplayName', 'Рунге-кутта (h=0.05)');

xlabel('x');

ylabel('Абсолютная погрешность');

title('Поведение абсолютной погрешности решения по всему интервалу');

legend();

grid on;

function [x, y] = eulerMethod(f, x0, y0, interval, h) % Метод Эйлера

% Инициализация

x = interval(1):h:interval(2);

n = length(x);

y = zeros(1, n);

y(1) = y0;

% Метод Эйлера

for i = 1:n-1

y(i+1) = y(i) + h \* f(x(i), y(i));

end

end

function [x, y] = rungeKutta4(f, x0, y0, interval, h) % Метод Рунге-Кутты

% Инициализация

x = interval(1):h:interval(2);

n = length(x);

y = zeros(1, n);

y(1) = y0;

% Метод Рунге-Кутта 4-го порядка

for i = 1:n-1

k1 = h \* f(x(i), y(i));

k2 = h \* f(x(i) + h/2, y(i) + k1/2);

k3 = h \* f(x(i) + h/2, y(i) + k2/2);

k4 = h \* f(x(i) + h, y(i) + k3);

y(i+1) = y(i) + (k1 + 2\*k2 + 2\*k3 + k4) / 6;

end

end

**Результаты расчётов (task1.m)**

Первая функция не является решением уравнения.

-x^3\*(x + x/log(x) + (2\*x\*(log(x) - 1))/log(x) - (x\*(log(x) - 1))/log(x)^2) == (x^4\*(log(x) - 1)^2)/log(x)^2

symbolic function inputs: x

Вторая функция является решением уравнения.

x^4 == x^4

symbolic function inputs: x

**Результаты расчётов (task2.m)**

===========================================================================

Погрешности по методу Рунге:

Метод Эйлера (h=0.2): 0.080365

Метод Эйлера (h=0.05): 0.019176

MATLAB ODE45: 0.005822

Метод Рунге-Кутта 4 порядка (h=0.2): 1.533980

Метод Рунге-Кутта 4 порядка (h=0.05): 0.543756

===========================================================================

Абсолютные погрешности в конце интервала:

Метод Эйлера (h=0.2): 0.080348

Метод Эйлера (h=0.05): 0.019176

Метод Рунге-Кутта (h=0.02): 0.000016

Метод Рунге-Кутта (h=0.005): 0.000000

===========================================================================



