Санкт-Петербургский политехнический университет Петра Великого

Институт компьютерных наук и технологий

Высшая школа программной инженерии

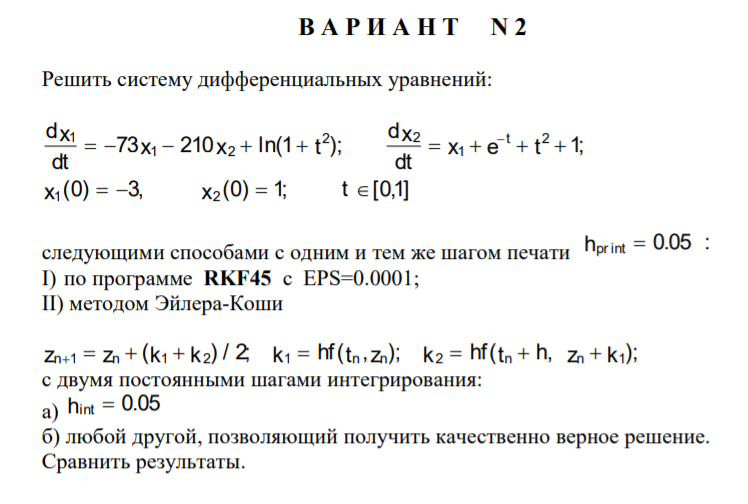
О Т Ч Е Т

Лабораторная работа № 3

по дисциплине «Вычислительная математика»

Выполнила студент гр. 3530904/90002 Афанасьев Е. Д.

Преподаватель Воскобойников С. П.

**Постанова задачи**

**Код программы**

#include <iostream>

#include <cmath>

#include <vector>

#include <array>

#include "cmath.h"

#include "rkf45.c"

const int n = 2;

const int COUNT\_PRINT = 21;

const double abserr = 10e-5;

const double H\_PRINT = 0.05;

const std::array<double, n> x\_0 = {-3, 1};

int func(int n, double t, double x[], double xp[]);

void rkf2(int (\*func)(int n, double t, double x[], double xp[]), double f[], double t, double h);

void do\_rkf2(double h, std::array<double, n>& arr, std::vector<std::array<double, n>>& res);

int main()

{

int fail = 1;

rkfinit(n, &fail);

std::array<double, n> f{x\_0[0], x\_0[1]};

std::array<double, n> f\_cond\_step{x\_0[0], x\_0[1]};

std::array<double, n> f\_stable\_step{x\_0[0], x\_0[1]};

std::array<double, n> fp{};

std::vector<std::array<double, n>> result;

std::vector<std::array<double, n>> result\_cond\_step;

std::vector<std::array<double, n>> result\_stable\_step;

double relerr = 10e-4, t = 0, t\_out = 0, h = 0;

int nfe = 0, max\_nfe = 10e4, l\_flag = 1;

for (double h\_print = 0; h\_print < 1 + H\_PRINT; h\_print += H\_PRINT)

{

rkf45(func, n, f.data(), fp.data(), &t, t\_out, &relerr, abserr, &h, &nfe, max\_nfe, &l\_flag);

t\_out += H\_PRINT;

result.push\_back(f);

}

h = 0.05;

do\_rkf2(h, f\_cond\_step, result\_cond\_step);

h = 0.001;

do\_rkf2(h, f\_stable\_step, result\_stable\_step);

int i0 = 0;

std::cout << "RKF45\n";

for (double hPrint = 0; hPrint < 1 + H\_PRINT; hPrint += H\_PRINT) {

std::cout << "X1: " << result[i0 \* result.size() / COUNT\_PRINT][0]

<< " X2: " << result[i0 \* result.size() / COUNT\_PRINT][1] << '\n';

i0++;

}

i0 = 0;

std::cout << "\nh = 0.05\n";

for (double hPrint = 0; hPrint < 1 + H\_PRINT; hPrint += H\_PRINT) {

std::cout << "X1: " << result\_cond\_step[i0 \* result\_cond\_step.size() / COUNT\_PRINT][0]

<< " X2: " << result\_cond\_step[i0 \* result\_cond\_step.size() / COUNT\_PRINT][1] << '\n';

i0++;

}

i0 = 0;

std::cout << "\nh = 0.001\n";

for (double hPrint = 0; hPrint < 1 + H\_PRINT; hPrint += H\_PRINT) {

std::cout << "X1: " << result\_stable\_step[i0 \* result\_stable\_step.size() / COUNT\_PRINT][0]

<< " X2: " << result\_stable\_step[i0 \* result\_stable\_step.size() / COUNT\_PRINT][1] << '\n';

i0++;

}

std::cout << "\ndifference\n";

i0 = 0;

for (double hPrint = 0; hPrint < 1 + H\_PRINT; hPrint += H\_PRINT) {

std::cout << "diff X1: " << std::abs(result\_stable\_step[i0 \* result\_stable\_step.size() / COUNT\_PRINT][0]

- result[i0 \* result.size() / COUNT\_PRINT][0])

<< " diff X2: " << std::abs(result\_stable\_step[i0 \* result\_stable\_step.size() / COUNT\_PRINT][1]

- result[i0 \* result.size() / COUNT\_PRINT][1]) << '\n';

i0++;

}

}

int func(int n, double t, double x[], double xp[])

{

xp[0] = -73 \* x[0] + -210 \* x[1] + std::log1p(t \* t);

xp[1] = x[0] + exp(-t) + t \* t + 1;

return 0;

}

void rkf2(int (\*func)(int n, double t, double x[], double xp[]), double f[], double t, double h)

{

std::array<double, n> temp{0, 0};

std::array<double, n> k\_1{0, 0};

std::array<double, n> k\_2{0, 0};

std::array<double, n> res\_1{0, 0};

std::array<double, n> res\_2{0, 0};

func(n, t, f, res\_1.data());

for (int i = 0; i < 2; i++)

{

k\_1[i] = h \* res\_1[i];

temp[i] = f[i] + k\_1[i];

}

func(n, t + h, temp.data(), res\_2.data());

for (int i = 0; i < 2; i++)

{

k\_2[i] = h \* res\_2[i];

f[i] += (k\_1[i] + k\_2[i]) / 2;

}

}

void do\_rkf2(double h, std::array<double, n>& arr, std::vector<std::array<double, n>>& res)

{

double t\_out = 0;

for (double h\_print = 0; h\_print < 1 + H\_PRINT; h\_print += h)

{

rkf2(func, arr.data(), t\_out, h);

res.push\_back(arr);

t\_out += h;

}

}

**Результат работы**

RKF45

X1: -3 X2: 1

X1: -2.78428 X2: 0.955328

X1: -2.66501 X2: 0.915858

X1: -2.55931 X2: 0.880217

X1: -2.46418 X2: 0.848192

X1: -2.379 X2: 0.819626

X1: -2.30346 X2: 0.794379

X1: -2.23712 X2: 0.772324

X1: -2.1797 X2: 0.753351

X1: -2.13087 X2: 0.737356

X1: -2.09039 X2: 0.724249

X1: -2.05799 X2: 0.713946

X1: -2.03349 X2: 0.706372

X1: -2.01665 X2: 0.701458

X1: -2.00733 X2: 0.699142

X1: -2.00533 X2: 0.699367

X1: -2.01052 X2: 0.70208

X1: -2.02276 X2: 0.707234

X1: -2.04192 X2: 0.714783

X1: -2.06788 X2: 0.724686

X1: -2.10053 X2: 0.736908

h = 0.05

X1: -3.10869 X2: 0.960093

X1: -3.84966 X2: 0.933019

X1: -6.85186 X2: 0.941864

X1: -18.0219 X2: 1.07084

X1: -58.7723 X2: 1.6257

X1: -206.725 X2: 3.71519

X1: -743.262 X2: 11.3589

X1: -2688.39 X2: 39.1284

X1: -9739.64 X2: 139.845

X1: -35300.6 X2: 504.989

X1: -127959 X2: 1828.67

X1: -463846 X2: 6627.05

X1: -1.68144e+06 X2: 24021.2

X1: -6.0952e+06 X2: 87075

X1: -2.20951e+07 X2: 315645

X1: -8.00947e+07 X2: 1.14421e+06

X1: -2.90343e+08 X2: 4.14776e+06

X1: -1.05249e+09 X2: 1.50356e+07

X1: -3.81529e+09 X2: 5.45042e+07

X1: -1.38304e+10 X2: 1.97578e+08

X1: -5.01353e+10 X2: 7.16219e+08

h = 0.001

X1: -2.99122 X2: 0.999004

X1: -2.78167 X2: 0.954499

X1: -2.6628 X2: 0.915109

X1: -2.55731 X2: 0.879541

X1: -2.46235 X2: 0.847587

X1: -2.37739 X2: 0.819089

X1: -2.30202 X2: 0.793907

X1: -2.23588 X2: 0.771915

X1: -2.17862 X2: 0.753002

X1: -2.12997 X2: 0.737066

X1: -2.08964 X2: 0.724015

X1: -2.05741 X2: 0.713768

X1: -2.03306 X2: 0.706248

X1: -2.01638 X2: 0.701387

X1: -2.0072 X2: 0.699122

X1: -2.00535 X2: 0.699397

X1: -2.01069 X2: 0.70216

X1: -2.02307 X2: 0.707361

X1: -2.04236 X2: 0.714958

X1: -2.06846 X2: 0.724908

X1: -2.10124 X2: 0.737175

difference between solutions

diff X1: 0.0087765 diff X2: 0.000995999

diff X1: 0.00260941 diff X2: 0.000829357

diff X1: 0.00220772 diff X2: 0.000749104

diff X1: 0.00200383 diff X2: 0.000675253

diff X1: 0.00182748 diff X2: 0.000604975

diff X1: 0.00161125 diff X2: 0.000536966

diff X1: 0.00143803 diff X2: 0.000472112

diff X1: 0.00124488 diff X2: 0.000409251

diff X1: 0.00107842 diff X2: 0.00034882

diff X1: 0.000901987 diff X2: 0.000290096

diff X1: 0.000742483 diff X2: 0.000233284

diff X1: 0.000578538 diff X2: 0.000177924

diff X1: 0.000425633 diff X2: 0.0001241

diff X1: 0.000271291 diff X2: 7.15092e-05

diff X1: 0.000124399 diff X2: 2.01715e-05

diff X1: 2.23526e-05 diff X2: 3.0115e-05

diff X1: 0.000163883 diff X2: 7.93644e-05

diff X1: 0.000304492 diff X2: 0.000127716

diff X1: 0.000441294 diff X2: 0.000175198

diff X1: 0.000576834 diff X2: 0.000221909

diff X1: 0.000709488 diff X2: 0.000267884

**Вывод**

Исходная матрица имеет вид . Ее собственные значения -3 и -70. Для метода Рунге-Кутты работает следующее условие устойчивости: h|λkmax| < 2. h < 0.02857. При заданном шаге 0.05 решение неустойчиво. Нужно взять шаг, который будет давать точное, устойчивое решение.