**Министерство науки и высшего образования Российской Федерации**

федеральное государственное автономное образовательное учреждение

высшего образования

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ**

**ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Инженерная школа природных ресурсов

Направление подготовки Химическая технология

Отделение химической инженерии

**Численные методы Решения обыкновенных дифференциальных уравнений**

**Лабораторная работа по дисциплине «Углубленный курс информатики»**

Выполнил студент гр. 2Д93 И.В. Петришина

(Подпись)

15.05. 2020 г.

Отчет принят:

Преподаватель

доцент ОХИ ИШПР, к.т.н. В.А. Чузлов

(Подпись)

\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ 2020 г.

Томск 2020 г.

**Цель работы:** изучить численные методы решений обыкновенных дифференциальных уравнений.

**Теоретическая часть**

Любая физическая ситуация, где рассматривается степень изменения одной переменной по отношению к другой переменной, описывается дифференциальным уравнением.

Дифференциальные уравнения устанавливают связь между независимыми переменными, искомыми функциями и их производными. Если искомая функция зависит от одной переменной, то дифференциальное уравнение называется обыкновенным.

Метод Эйлера является сравнительно «грубым» методом решения дифференциальных уравнений, однако идеи, положенные в его основу, являются, по существу, исходными для очень широкого класса численных методов.

Метод Рунге-Кутта требует существенно большего объема вычислений по сравнению с методом Эйлера, однако это окупается повышенной точностью, что дает возможность проводить вычисления с бóльшим шагом. Другими словами, для получения результатов с одинаковой точностью в методе Эйлера потребуется значительно меньший шаг, чем в методе Рунге-Кутта.

**Практическая часть**

**Задание**

где 𝑦 ̃ - расчетное значение, - значение, полученное из аналитического решения.

|  |  |  |  |
| --- | --- | --- | --- |
| **Дифференциальное уравнение** | **Отрезок, шаг** | **Начальные условия** | **Аналитическое решение** |
|  | [1;2];  h = 0.1 | x0 = 1;  y0 =3.0307 |  |
|  | [1; 2];  h = 0.2 | x0 = 1;  y0 = e |  |
|  | [0; 2];  h = 0.2 | x0 = 0;  y0 = ln(2) |  |
|  | [2; 3];  h = 0.2 | x0 = 2;  y0 = 4 |  |

**Программная реализация, 1 уравнение**

**Метод Рунге-Кутты**

**program** lb12\_1;

**type**

matrix = **array of array of** real;

**function** f(x, y: real): real;

**begin**

result := -(2 \* y + 1) \* cos(x) / sin(x)

**end**;

**function** a\_resh(x: real): real;

**begin**

result := 5 / (2 \* sqr(sin(x))) - 1 / 2

**end**;

**function** runge\_kutt\_method(start\_x, stop\_x, h, start\_y: real): matrix;

**var**

x, y: real;

k1, k2, k3, k4, Ay: real;

i: integer;

**begin**

SetLength(result, Trunc((stop\_x - start\_x) / h) + 1);

**for** i := 0 **to** High(result) **do**

SetLength(result[i], 2);

x := start\_x;

y := start\_y;

**for** i := 0 **to** High(result) **do**

**begin**

result[i, 0] := x;

result[i, 1] := y;

k1 := f(x, y);

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

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

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

y := y + h / 6 \* (k1 + 2 \* k2 + 2 \* k3 + k4);

x := x + h;

**end**;

**end**;

**function** f\_er(y: matrix): matrix;

**var**

i, j: integer;

**begin**

SetLength(result, Length (y));

**for** i := 0 **to** High(result) **do**

SetLength(result[i], 3);

**for** i := 0 **to** High(result) **do**

**begin**

result[i, 2] := abs(y[i, 1] - a\_resh(y[i, 0]))/a\_resh(y[i, 0]) \* 100;

**for** j := 0 **to** 1 **do**

result [i, j] := y[i, j];

**end**;

**end**;

**procedure** print\_results(y: matrix);

**var**

i, j: integer;

**begin**

**for** i := 0 **to** High(y) **do**

**begin**

**for** j := 0 **to** High(y[i]) **do**

write(y[i, j]:10:4);

writeln

**end**

**end**;

**begin**

print\_results(f\_er(runge\_kutt\_method(1, 2, 0.1, 3.0307)));

**end**.

**Ответ**

1.0000 3.0307 0.0002

1.1000 2.6476 0.0003

1.2000 2.3779 0.0003

1.3000 2.1927 0.0003

1.4000 2.0744 0.0003

1.5000 2.0126 0.0003

1.6000 2.0021 0.0003

1.7000 2.0422 0.0003

1.8000 2.1361 0.0003

1.9000 2.2918 0.0003

2.0000 2.5236 0.0003

Погрешность минимальна, результаты аналитического решения близки к расчетным значениям.

**Метод Эйлера**

**program** lb12\_1;

**type**

matrix = **array of array of** real;

**function** f(x, y: real): real;

**begin**

result := -(2 \* y + 1) \* cos(x) / sin(x)

**end**;

**function** a\_resh(x: real): real;

**begin**

result := 5 / (2 \* sqr(sin(x))) - 1 / 2

**end**;

**function** eyler\_method(start\_x, stop\_x, h, start\_y: real): matrix;

**var**

x, y: real;

i: integer;

**begin**

SetLength(result, Trunc((stop\_x - start\_x) / h) + 1);

**for** i := 0 **to** High(result) **do**

SetLength(result[i], 2);

x := start\_x;

y := start\_y;

**for** i := 0 **to** High(result) **do**

**begin**

result[i, 0] := x;

result[i, 1] := y;

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

x := x + h

**end**;

**end**;

**function** f\_er(y: matrix): matrix;

**var**

i, j: integer;

**begin**

SetLength(result, Length(y));

**for** i := 0 **to** High(result) **do**

SetLength(result[i], 3);

**for** i := 0 **to** High(result) **do**

**begin**

result[i, 2] := abs(y[i, 1] - a\_resh(y[i, 0])) / a\_resh(y[i, 0]) \* 100;

**for** j := 0 **to** 1 **do**

result[i, j] := y[i, j];

**end**;

**end**;

**procedure** print\_results(y: matrix);

**var**

i, j: integer;

**begin**

**for** i := 0 **to** High(y) **do**

**begin**

**for** j := 0 **to** High(y[i]) **do**

write(y[i, j]:8:4);

writeln

**end**

**end**;

**begin**

print\_results(f\_er(eyler\_method(1, 2, 0.1, 3.0307)))

**end**.

**Ответ**

1.0000 3.0307 0.0002

1.1000 2.5773 2.6563

1.2000 2.2640 4.7870

1.3000 2.0491 6.5469

1.4000 1.9076 8.0402

1.5000 1.8245 9.3430

1.6000 1.7916 10.5170

1.7000 1.8050 11.6172

1.8000 1.8649 12.6973

1.9000 1.9752 13.8141

2.0000 2.1443 15.0302

Отклонения в измерениях есть. Результаты аналитического решения различаются с расчетными значениями.

**Программная реализация, 2 уравнение**

**Метод Рунге-Кутты**

**program** lb12\_2;

**type**

matrix = **array of array of** real;

**function** f(x, y: real): real;

**begin**

result := - (y\*ln(y))/x

**end**;

**function** a\_resh(x: real): real;

**begin**

result := exp((1/x)\*ln(e))

**end**;

**function** runge\_kutt\_method(start\_x, stop\_x, h, start\_y: real): matrix;

**var**

x, y: real;

k1, k2, k3, k4: real;

i: integer;

**begin**

SetLength(result, Trunc((stop\_x - start\_x) / h)+1);

**for** i := 0 **to** High(result) **do**

SetLength(result[i], 2);

x := start\_x;

y := start\_y;

**for** i := 0 **to** High(result) **do**

**begin**

result[i, 0] := x;

result[i, 1] := y;

k1 := f(x, y);

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

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

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

y := y + h / 6 \* (k1 + 2 \* k2 + 2 \* k3 + k4);

x := x + h

**end**;

**end**;

**function** f\_er(y: matrix): matrix;

**var**

i, j: integer;

**begin**

SetLength(result, Length(y));

**for** i := 0 **to** High(result) **do**

SetLength(result[i], 3);

**for** i := 0 **to** High(result) **do**

**begin**

result[i, 2] := abs(y[i, 1] - a\_resh(y[i, 0])) / a\_resh(y[i, 0]) \* 100;

**for** j := 0 **to** 1 **do**

result[i, j] := y[i, j];

**end**;

**end**;

**procedure** print\_results(y: matrix);

**var**

i, j: integer;

**begin**

**for** i := 0 **to** High(y) **do**

**begin**

**for** j := 0 **to** High(y[i]) **do**

write(y[i, j]:10:4);

writeln

**end**

**end**;

**begin**

print\_results(f\_er(runge\_kutt\_method(1, 2, 0.2, e)));

**end**.

**Ответ**

1.0000 2.7183 0.0000

1.2000 2.3010 0.0003

1.4000 2.0427 0.0004

1.6000 1.8683 0.0004

1.8000 1.7429 0.0003

2.0000 1.6487 0.0003

Погрешность минимальна, результаты аналитического решения близки к расчетным значениям.

**Метод Эйлера**

**program** lb12\_2;

**type**

matrix = **array of array of** real;

**function** f(x, y: real): real;

**begin**

result := -(y \* ln(y)) / x

**end**;

**function** a\_resh(x: real): real;

**begin**

result := exp((1 / x) \* ln(e))

**end**;

**function** eyler\_method(start\_x, stop\_x, h, start\_y: real): matrix;

**var**

x, y: real;

i: integer;

**begin**

SetLength(result, Trunc((stop\_x - start\_x) / h) + 1);

**for** i := 0 **to** High(result) **do**

SetLength(result[i], 2);

x := start\_x;

y := start\_y;

**for** i := 0 **to** High(result) **do**

**begin**

result[i, 0] := x;

result[i, 1] := y;

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

x := x + h

**end**;

**end**;

**function** f\_er(y: matrix): matrix;

**var**

i, j: integer;

**begin**

SetLength(result, Length(y));

**for** i := 0 **to** High(result) **do**

SetLength(result[i], 3);

**for** i := 0 **to** High(result) **do**

**begin**

result[i, 2] := abs(y[i, 1] - a\_resh(y[i, 0])) / a\_resh(y[i, 0]) \* 100;

**for** j := 0 **to** 1 **do**

result[i, j] := y[i, j];

**end**;

**end**;

**procedure** print\_results(y: matrix);

**var**

i, j: integer;

**begin**

**for** i := 0 **to** High(y) **do**

**begin**

**for** j := 0 **to** High(y[i]) **do**

write(y[i, j]:8:4);

writeln

**end**

**end**;

**begin**

print\_results(f\_er(eyler\_method(1, 2, 0.2, e)))

**end**.

**Ответ**

1.0000 2.7183 0.0000

1.2000 2.1746 5.4912

1.4000 1.8931 7.3267

1.6000 1.7205 7.9098

1.8000 1.6038 7.9825

2.0000 1.5196 7.8312

Отклонения в измерениях есть. Результаты аналитического решения различаются с расчетными значениями.

**Программная реализация, 3 уравнение**

**Метод Рунге-Кутты**

**program** lb12\_3;

**type**

matrix = **array of array of** real;

**function** f(x, y: real): real;

**begin**

result := 2 \* x / exp((y - sqr(x)) \* ln(e))

**end**;

**function** a\_resh(x: real): real;

**begin**

result := ln((exp(sqr(x)) \* ln(e))+1)

**end**;

**function** runge\_kutt\_method(start\_x, stop\_x, h, start\_y: real): matrix;

**var**

x, y: real;

k1, k2, k3, k4: real;

i: integer;

**begin**

SetLength(result, Trunc((stop\_x - start\_x) / h) + 1);

**for** i := 0 **to** High(result) **do**

SetLength(result[i], 2);

x := start\_x;

y := start\_y;

**for** i := 0 **to** High(result) **do**

**begin**

result[i, 0] := x;

result[i, 1] := y;

k1 := f(x, y);

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

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

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

y := y + h / 6 \* (k1 + 2 \* k2 + 2 \* k3 + k4);

x := x + h

**end**;

**end**;

**function** f\_er(y: matrix): matrix;

**var**

i, j: integer;

**begin**

SetLength(result, Length(y));

**for** i := 0 **to** High(result) **do**

SetLength(result[i], 3);

**for** i := 0 **to** High(result) **do**

**begin**

result[i, 2] := abs(y[i, 1] - a\_resh(y[i, 0])) / a\_resh(y[i, 0]) \* 100;

**for** j := 0 **to** 1 **do**

result[i, j] := y[i, j];

**end**;

**end**;

**procedure** print\_results(y: matrix);

**var**

i, j: integer;

**begin**

**for** i := 0 **to** High(y) **do**

**begin**

**for** j := 0 **to** High(y[i]) **do**

write(y[i, j]:10:4);

writeln

**end**

**end**;

**begin**

print\_results(f\_er(runge\_kutt\_method(0, 2, 0.2, ln(2))));

**end**.

**Ответ**

0.0000 0.6931 0.0000

0.2000 0.7133 0.0001

0.4000 0.7763 0.0006

0.6000 0.8893 0.0015

0.8000 1.0635 0.0031

1.0000 1.3133 0.0055

1.2000 1.6528 0.0089

1.4000 2.0920 0.0128

1.6000 2.6349 0.0164

1.8000 3.2790 0.0190

2.0000 4.0190 0.0204

Погрешность минимальна, результаты аналитического решения близки к расчетным значениям.

**Метод Эйлера**

**program** lb12\_3;

**type**

matrix = **array of array of** real;

**function** f(x, y: real): real;

**begin**

result := 2\*x/exp((y-sqr(x))\*ln(e))

**end**;

**function** a\_resh(x: real): real;

**begin**

result := ln((exp(sqr(x)) \* ln(e)) + 1)

**end**;

**function** eyler\_method(start\_x, stop\_x, h, start\_y: real): matrix;

**var**

x, y: real;

i: integer;

**begin**

SetLength(result, Trunc((stop\_x - start\_x) / h)+1);

**for** i := 0 **to** High(result) **do**

SetLength(result[i], 2);

x := start\_x;

y := start\_y;

**for** i := 0 **to** High(result) **do**

**begin**

result[i, 0] := x;

result[i, 1] := y;

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

x := x + h

**end**;

**end**;

**function** f\_er(y: matrix): matrix;

**var**

i, j: integer;

**begin**

SetLength(result, Length(y));

**for** i := 0 **to** High(result) **do**

SetLength(result[i], 3);

**for** i := 0 **to** High(result) **do**

**begin**

result[i, 2] := abs(y[i, 1] - a\_resh(y[i, 0])) / a\_resh(y[i, 0]) \* 100;

**for** j := 0 **to** 1 **do**

result[i, j] := y[i, j];

**end**;

**end**;

**procedure** print\_results(y: matrix);

**var**

i, j: integer;

**begin**

**for** i := 0 **to** High(y) **do**

**begin**

**for** j := 0 **to** High(y[i]) **do**

write(y[i, j]:8:4);

writeln

**end**

**end**;

**begin**

print\_results(f\_er(eyler\_method(0, 2, 0.2, ln(2))))

**end**.

**Ответ**

0.0000 0.6931 0.0000

0.2000 0.6931 2.8317

0.4000 0.7348 5.3538

0.6000 0.8248 7.2451

0.8000 0.9756 8.2639

1.0000 1.2044 8.2910

1.2000 1.5304 7.3937

1.4000 1.9689 5.8728

1.6000 2.5240 4.1947

1.8000 3.1874 2.7749

2.0000 3.9463 1.7882

Отклонения в измерениях есть. Результаты аналитического решения различаются с расчетными значениями.

**Программная реализация, 4 уравнение**

**Метод Рунге-Кутты**

**program** lb12\_4;

**type**

matrix = **array of array of** real;

**function** f(x, y: real): real;

**begin**

result := (y-3)/(3\*sqr(x)+x)

**end**;

**function** a\_resh(x: real): real;

**begin**

result := 3.5\*x/(3\*x + 1) +3

**end**;

**function** runge\_kutt\_method(start\_x, stop\_x, h, start\_y: real): matrix;

**var**

x, y: real;

k1, k2, k3, k4: real;

i: integer;

**begin**

SetLength(result, Trunc((stop\_x - start\_x) / h)+1);

**for** i := 0 **to** High(result) **do**

SetLength(result[i], 2);

x := start\_x;

y := start\_y;

**for** i := 0 **to** High(result) **do**

**begin**

result[i, 0] := x;

result[i, 1] := y;

k1 := f(x, y);

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

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

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

y := y + h / 6 \* (k1 + 2 \* k2 + 2 \* k3 + k4);

x := x + h

**end**;

**end**;

**function** f\_er(y: matrix): matrix;

**var**

i, j: integer;

**begin**

SetLength(result, Length(y));

**for** i := 0 **to** High(result) **do**

SetLength(result[i], 3);

**for** i := 0 **to** High(result) **do**

**begin**

result[i, 2] := abs(y[i, 1] - a\_resh(y[i, 0])) / a\_resh(y[i, 0]) \* 100;

**for** j := 0 **to** 1 **do**

result[i, j] := y[i, j];

**end**;

**end**;

**procedure** print\_results(y: matrix);

**var**

i, j: integer;

**begin**

**for** i := 0 **to** High(y) **do**

**begin**

**for** j := 0 **to** High(y[i]) **do**

write(y[i, j]:10:4);

writeln

**end**

**end**;

**begin**

print\_results(f\_er(runge\_kutt\_method(2, 3, 0.2, 4)));

**end**.

**Ответ**

2.0000 4.0000 0.0000

2.2000 4.0132 0.0000

2.4000 4.0244 0.0000

2.6000 4.0341 0.0000

2.8000 4.0426 0.0000

3.0000 4.0500 0.0000

Погрешность отсутствует, результаты аналитического решения совпали к расчетным значениям

**Метод Эйлера**

**program** lb12\_4;

**type**

matrix = **array of array of** real;

**function** f(x, y: real): real;

**begin**

result := (y - 3) / (3 \* sqr(x) + x)

**end**;

**function** a\_resh(x: real): real;

**begin**

result := 3.5 \* x / (3 \* x + 1) + 3

**end**;

**function** eyler\_method(start\_x, stop\_x, h, start\_y: real): matrix;

**var**

x, y: real;

i: integer;

**begin**

SetLength(result, Trunc((stop\_x - start\_x) / h) + 1);

**for** i := 0 **to** High(result) **do**

SetLength(result[i], 2);

x := start\_x;

y := start\_y;

**for** i := 0 **to** High(result) **do**

**begin**

result[i, 0] := x;

result[i, 1] := y;

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

x := x + h

**end**;

**end**;

**function** f\_er(y: matrix): matrix;

**var**

i, j: integer;

**begin**

SetLength(result, Length(y));

**for** i := 0 **to** High(result) **do**

SetLength(result[i], 3);

**for** i := 0 **to** High(result) **do**

**begin**

result[i, 2] := abs(y[i, 1] - a\_resh(y[i, 0])) / a\_resh(y[i, 0]) \* 100;

**for** j := 0 **to** 1 **do**

result[i, j] := y[i, j];

**end**;

**end**;

**procedure** print\_results(y: matrix);

**var**

i, j: integer;

**begin**

**for** i := 0 **to** High(y) **do**

**begin**

**for** j := 0 **to** High(y[i]) **do**

write(y[i, j]:8:4);

writeln

**end**

**end**;

**begin**

print\_results(f\_er(eyler\_method(2, 3, 0.2, 4)))

**end**.

**Ответ**

2.0000 4.0000 0.0000

2.2000 4.0143 0.0281

2.4000 4.0264 0.0504

2.6000 4.0368 0.0684

2.8000 4.0459 0.0831

3.0000 4.0539 0.0953

Погрешность минимальна, результаты аналитического решения близки к расчетным значениям

**Выводы**

В ходе лабораторной работы были изучены численные методы решений обыкновенных дифференциальных уравнений. Также, определена погрешность расчетного значения переменной y для каждого из методов вычисления.