Университет ИТМО

Кафедра ВТ

Лабораторная работа №4

по Вычислительной математике

***Решение ОДУ***

Выполнил: Татаринов Данил

Группа: P3211

Преподаватель: Петрова Мария Максимовна

Санкт-Петербург

2017

# Пояснение метода

Рассмотрим задачу Коши для системы обыкновенных дифференциальных уравнений первого порядка.

\textbf{y}'=\textbf{f}(x,\textbf{y}), \quad \textbf{y}(x_0)=\textbf{y}_0.

Тогда приближенное значение в последующих точках вычисляется по итерационной формуле:

 \textbf{y}_{n+1} = \textbf{y}_n + {h \over 6}(\textbf{k}_1 + 2\textbf{k}_2 + 2\textbf{k}_3 + \textbf{k}_4) 

Вычисление нового значения проходит в четыре стадии:

 \textbf{k}_1 = \textbf{f} \left( x_n, \textbf{y}_n \right), 

 \textbf{k}_2 = \textbf{f} \left( x_n + {h \over 2}, \textbf{y}_n + {h \over 2} \textbf{k}_1 \right), 

 \textbf{k}_3 = \textbf{f} \left( x_n + {h \over 2}, \textbf{y}_n + {h \over 2} \textbf{k}_2 \right), 

 \textbf{k}_4 = \textbf{f} \left( x_n + h, \textbf{y}_n + h\ \textbf{k}_3 \right). 

где h — величина шага сетки по x.

# Класс Difur

# public class Difur{

# private Function func;

# private double x0;

# private double y0;

# private double x1;

# private double acc;

# private double[] x;

# private double[] y;

# private int steps;

# public Difur(double x0, double y0, double x1, double acc, Function func){

# this.x0 = x0;

# this.y0 = y0;

# this.x1 = x1;

# this.acc = acc;

# this.func = func;

# }

# public void RungKut(){

# double k1, k2, k3, k4;

# steps = (int)((x1-x0)/acc);

# x = new double[steps+1];

# y = new double[steps+1];

# y[0] = y0;

# x[0] = x0;

# double y\_prev = y[0];

# for(int i=1;i<=steps;i++){

# k1 = func.calculate(i,y\_prev);

# k2 = func.calculate(i+acc/2,y\_prev+(k1\*acc/2));

# k3 = func.calculate(i+acc/2,y\_prev+(k2\*acc/2));

# k4 = func.calculate(i+acc,y\_prev+k3\*acc);

# y[i] = y\_prev+acc\*((k1+k2\*2+k3\*2+k4)/6);

# x[i] = x0+acc\*i;

# y\_prev = y[i];

# }

# }

# public double[] getX(){

# return x;

# }

# public double[] getY() {

# return y;

# }

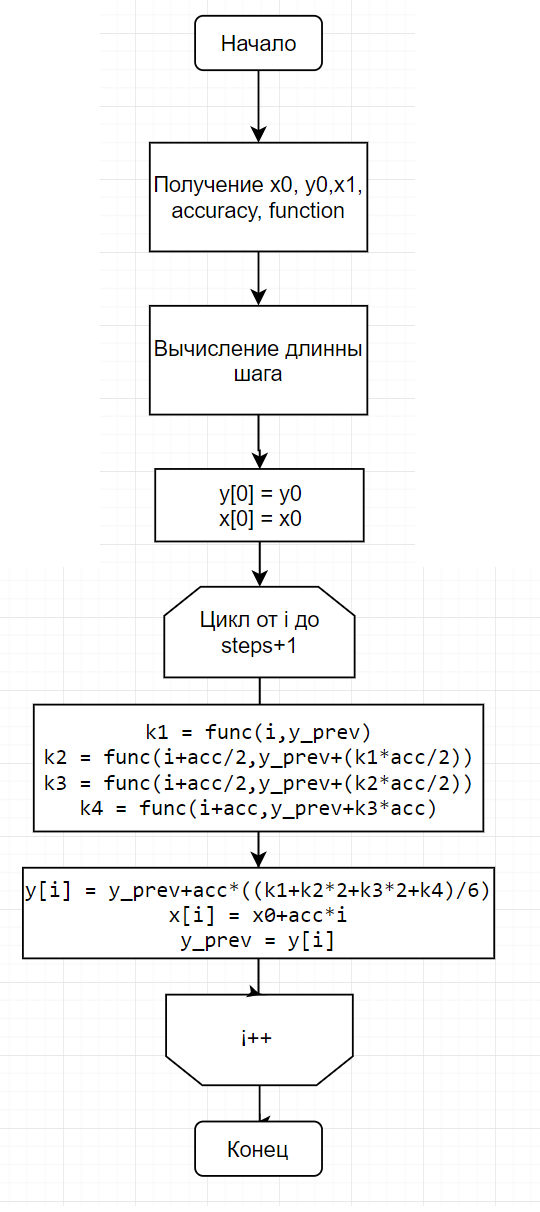
# public int getSteps(){

# return steps;

# }

}

# Блок схема метода с решением



# Тестовые функции

1);2); 3)

# Вывод

Научился решать ОДУ методом Рунге - Кутты.

Попрактиковался в использовании javafx и рисовании графиков посредством LineChart.