Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«Национальный исследовательский университет ИТМО»

Факультет Программной Инженерии и Компьютерной Техники

Лабораторная работа № 5

по дисциплине «Вычислительная математика»

Численное дифференцирование и задача Коши

Выполнил:

Гурьянов Кирилл Алексеевич

Группа: P32302

Преподаватель:

Перл Ольга Вячеславовна

Санкт-Петербург

2023

1. **Описание методов, расчетные формулы. Метод Эйлера.**

Метод Эйлера — это простейший численный метод для решения задачи Коши, которая заключается в решении дифференциального уравнения, удовлетворяющего начальным условиям.

Метод Эйлера заключается в приближенном нахождении значений функции y на отрезке. Метод Эйлера заменяет искомую функцию ломанной, состоящей из отрезков касательных к функции в точках x_i.

Для этого в начальной точке строим касательную к интегральной кривой:

$y - y_0 = y'_0\ (x - x_0)$, где $y'_0 = f(x_0, y_0)$

Находим координату следующих точек по формуле:

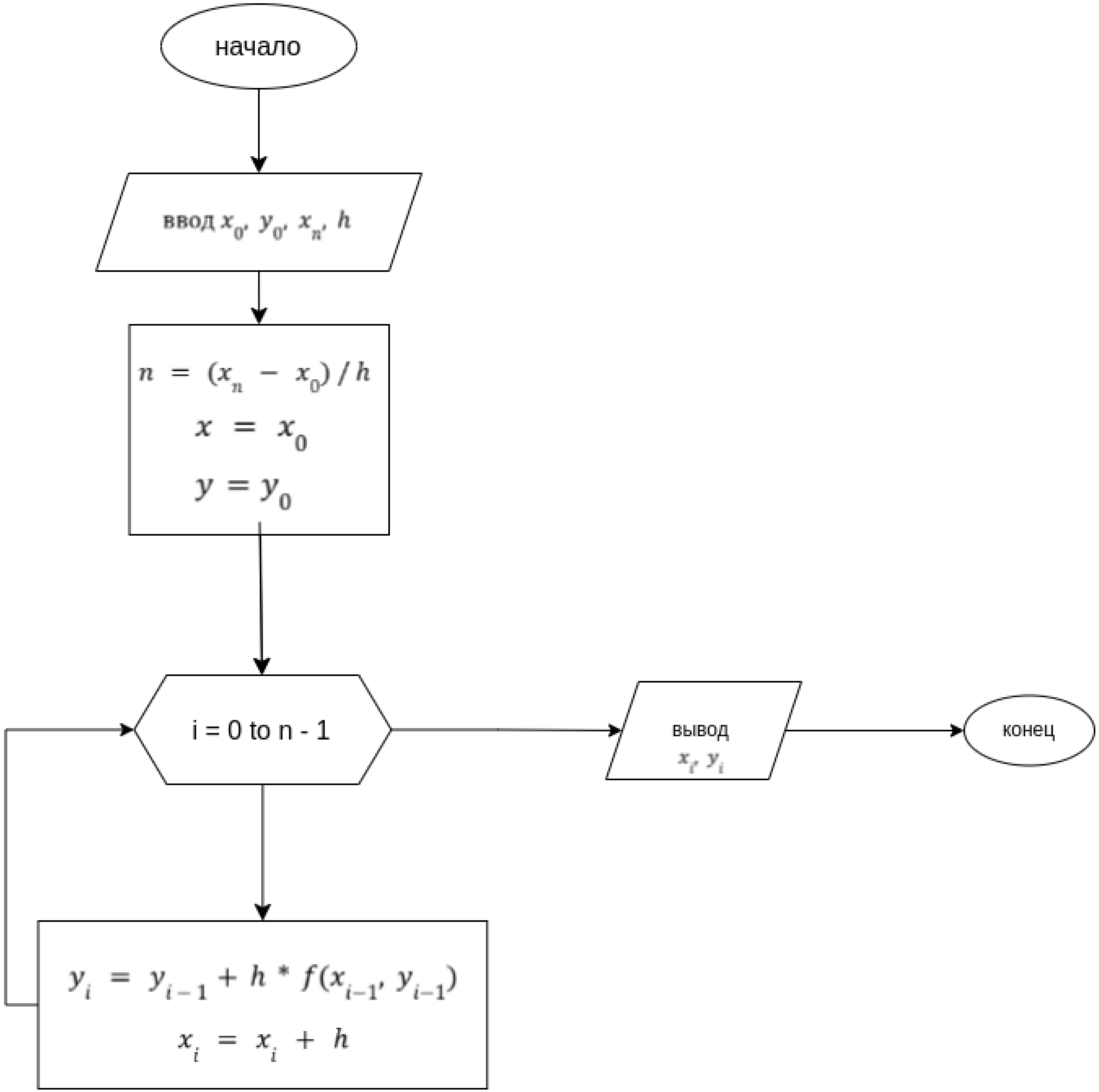
$y_i = y_{i-1} + h * f(x_{i-1}, y_{i-1})$, где $h = (x_i - x_{i-1})$

Шаг h должен быть достаточно малым, чтобы обеспечить достаточную точность решения, но не слишком малым, чтобы не привести к существенному увеличению времени вычислений.

Стоит отметить, что уменьшение шага в n раз приводит к уменьшению погрешности примерно в такое же количество раз.

Алгоритмическая сложность метода — O(n), где $n = (x_n - x_0)/h$

2. **Блок-схемы методов**

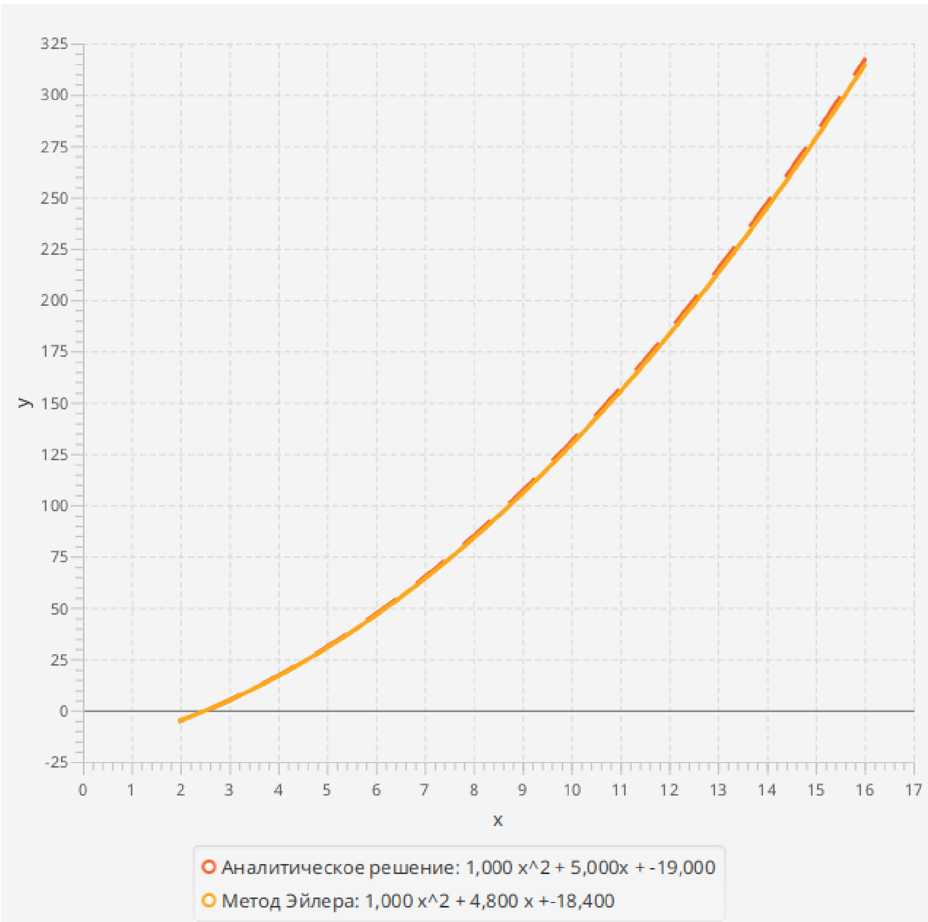


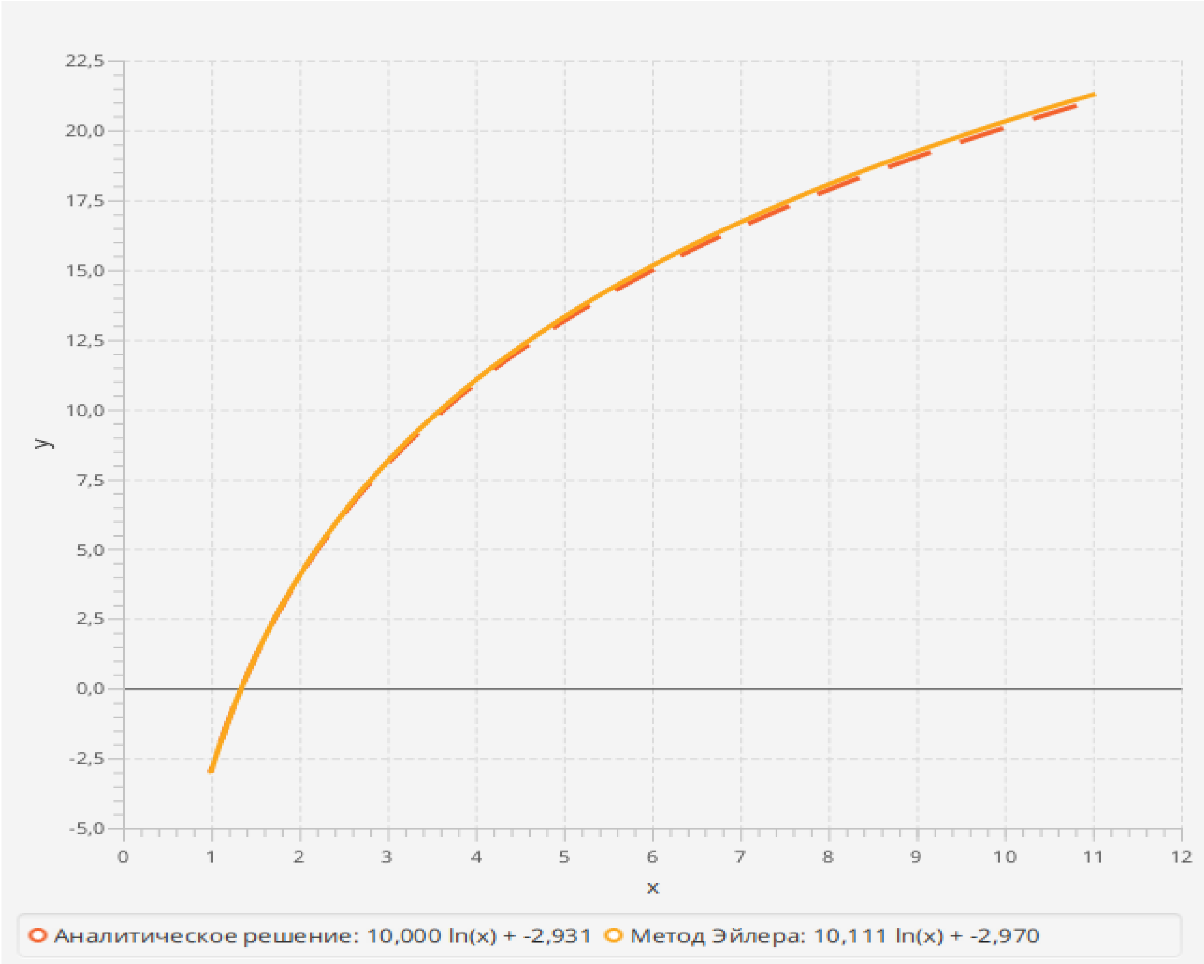
начало
ввод x_0, y_0, x_n, h
$n = (x_n - x_0) / h$
$x = x_0$
$y = y_0$
i = 0 to n - 1
вывод x_i, y_i
конец
$y_i = y_{i-1} + h * f(x_{i-1}, y_{i-1})$
$x_i = x_i + h$

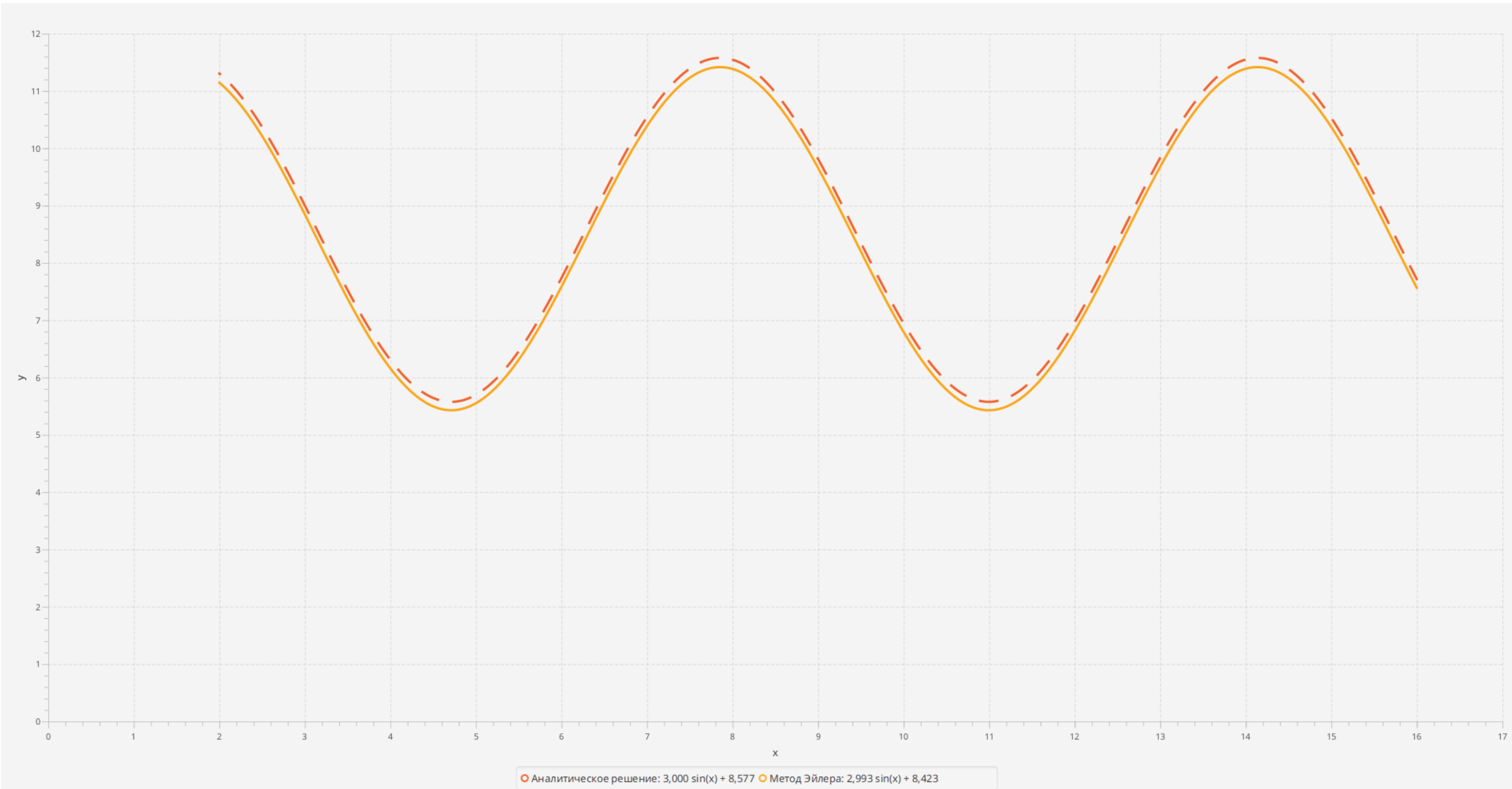
3. Листинг реализованных методов программы

```java
package lab5.algorithms;

import lab5.data.AnswerFromEulerMethod;
import lab5.data.DifferentialEquation;

3 usages
public class EulerMethod {
    1 usage
    public AnswerFromEulerMethod solveEquation(DifferentialEquation differentialEquation, double x0, double y0, double h, double lastX) {
        int numberOfSteps = (int) ((lastX - x0) / h);
        double[] xArray = new double[numberOfSteps + 1];
        double[] yArray = new double[numberOfSteps + 1];
        xArray[0] = x0;
        yArray[0] = y0;
        double x = x0;
        double y = y0;
        for (int i = 0; i < numberOfSteps; i++) {
            y = y + h * differentialEquation.calculateFunction(x, y);
            x += h;
            xArray[i + 1] = x;
            yArray[i + 1] = y;
        }

        return new AnswerFromEulerMethod(xArray, yArray);
    }
}
```

4. Примеры и результаты работы программы на разных данных




y
x
Аналитическое решение: 10,000 ln(x) + -2,931
Метод Эйлера: 10,111 ln(x) + -2,970


y
x
Аналитическое решение: 3,000 sin(x) + 8,577
Метод Эйлера: 2,993 sin(x) + 8,423

5. Вывод

Для решения задачи Коши могут использоваться одношаговые и многошаговые методы. Метод Эйлера, усовершенствованный метод Эйлера и метод Рунге-Кутты являются одношаговыми. Такие методы для для вычисления каждой следующей точки искомой функции используют лишь одну предыдущую точку.

Среди одношаговых методов метод Эйлера является самым простым методом, он требует произведение наименьшего количества вычислений. К примеру, в усовершенствованном методе Эйлера для вычисления каждой следующей точки необходимо произвести вычисление значения производной в двух точках, а в методе Рунге-Кутты — 4 раза. Метод Эйлера же для вычисления каждой точки считает значение производной лишь один раз. Однако он является неустойчивым, имеет тенденцию к накоплению ошибок, т.е. чем больше отрезок, на котором необходимо найти искомую функцию, тем данный метод будет иметь большую погрешность при приближении к правой границе отрезка.

Усовершенствованный метод Эйлера требует большее количество вычислений, но благодаря этому имеет значительно большую точность. Метод Рунге-Кутты из описанных одношаговых методов является самым устойчивым и имеет наименьшую погрешность вычислений.

Порядок точности метода Эйлера — $O(h)$, т.е. при уменьшении шага в n раз данный метод будет выдавать примерно в n раз меньшую погрешность. Порядок точности усовершенствованного метода Эйлера – $O(h^2)$, а метода Рунге-Кутты — $O(h^4)$.

Многошаговые методы, в отличие от одношаговых, для нахождения каждой следующей точки используют для вычислений несколько значений, полученных на предыдущих шагах. Для вычисления начальных точек многошаговые методы используют одношаговые. Таким образом метод Милна использует для вычисления значения на текущем шаге два предыдущих значения, метод Адамса может быть разных порядком и использовать 2, 3 и 4 точки с предыдущих шагов. Однако именно методом Адамса принято называть метод четвертого порядка, т.е. использующих значения с предыдущих четырех шагов.