

Санкт-Петербургский *Национальный Исследовательский*  
*Университет ИТМО*  
Факультет программной инженерии и компьютерной техники

Вычислительная математика

Лабораторная работа №2

Вариант: 2аб

Выполнил:

студент группы Р3231

Нестеров Иван Алексеевич

Преподаватель:

Перл Ольга Вячеславовна

г. Санкт-Петербург

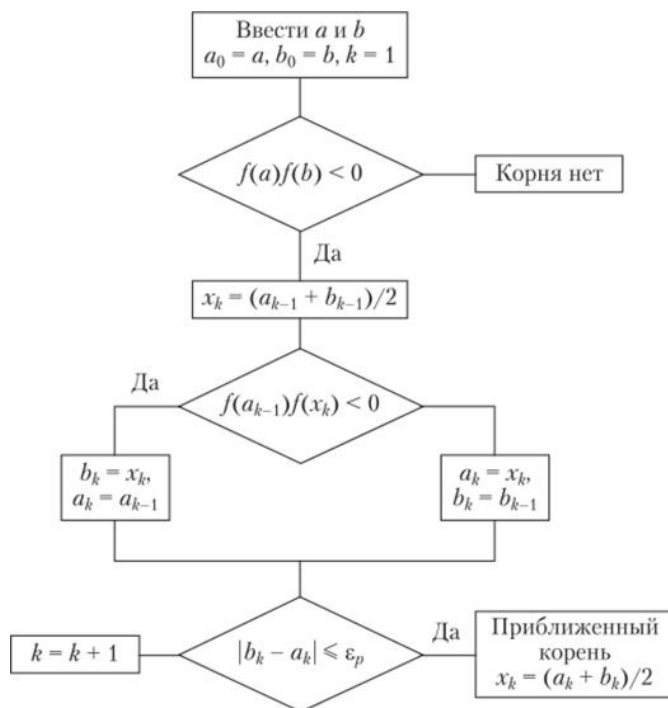
2022 г.

## Цель работы:

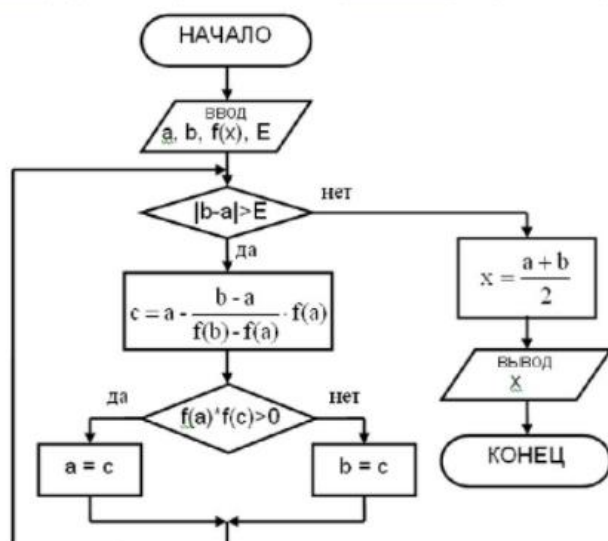
1. Реализовать алгоритмы решений нелинейных уравнений методом деления пополам и методом хорд;
2. Реализовать алгоритм решения систем нелинейных уравнений методом простых итераций

Блок-схемы указанных методов:

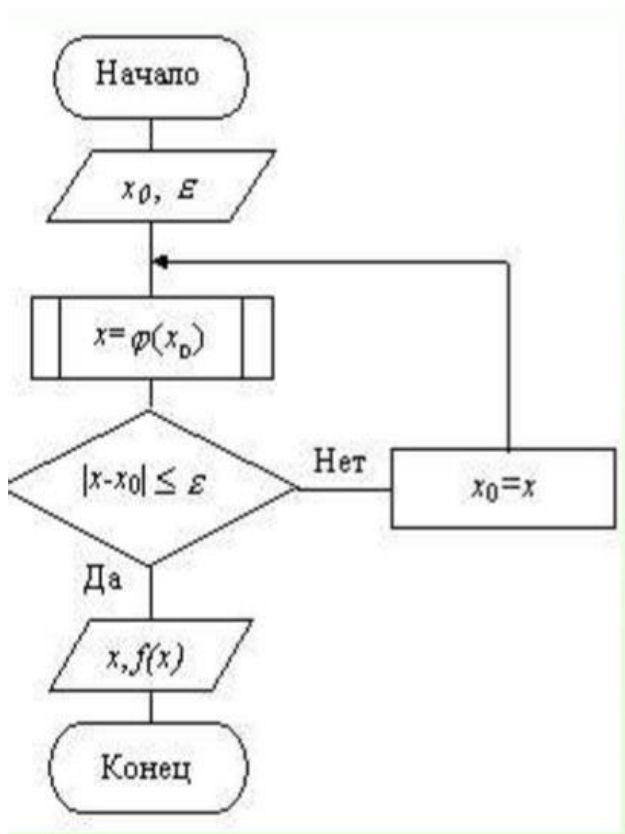
### 1. Метод деления пополам:



### 2. Метод хорд:



### 3. Метод простых итераций:



#### Описание методов:

1. Метод деления пополам. Суть метода в разбиении исследуемого отрезка  $[a;b]$  пополам, при условии, что произведение значений функций в данных точках  $a$  и  $b$  отрицательно, определение знака функции в середине отрезка  $((a+b)/2)$  и дальнейшем выборе той половины отрезка, где функция меняет знак, и, соответственно, содержит решение. Повторение действий для этой половины отрезка. Деление отрезка продолжается до тех пор, пока не будет достигнута указанная точность  $\varepsilon$ .
2. Метод хорд. Похож на предыдущий, однако деление отрезка проводится не в его середине, как у метода деления пополам, а в точке пересечения хорды с осью абсцисс, где хорда – отрезок, соединяющий точки  $a, b$ . За счет более «удачного» деления находит корень быстрее, чем предыдущий. Разумеется, если отрезок  $[a;b]$  и  $\varepsilon$  для работы двух этих методов был задан один и тот же.
3. Метод простых итераций. Используется для решения СНАУ вида  $Ax = B$ . Суть метода же – в нахождении нового значения  $x$  через старое. Предполагается некоторое значение, находится корень с большой погрешностью, далее вновь находится корень, но уже с использованием нового значения. Действие повторяется, пока точность не станет

достаточно мала. Однако решения может не найтись вовсе, поэтому разумно указать максимальное количество итераций, за которое алгоритм попытается найти решение.

Листинги вычислительных методов:

### 1. Метод деления пополам:

```
public void solveEquation() {
    DataReceiver receiver = new DataReceiver();
    double[] coefficients = receiver.receiveCoefficients();
    double a = coefficients[0];
    double b = coefficients[1];
    double e = coefficients[2];
    int equationNumber = receiver.receiveEquation();
    if (equationNumber == 1) {
        while (true) {
            double x = (a + b) / 2;
            if (receiver.solveFirstEquation(a) * receiver.solveFirstEquation(x) < 0) {
                b = x;
            } else if (receiver.solveFirstEquation(x) * receiver.solveFirstEquation(b) < 0) {
                a = x;
            }
            if (Math.abs(b - a) < 2 * e) {
                continue;
            } else {
                x = (a + b) / 2;
                if (receiver.solveFirstEquation(x) == 0) {
                    System.out.println("x = " + x + ". Is exact value.");
                } else System.out.println("x = " + x + ". Is approximate value accurate to ε = " + e + ".");
                break;
            }
        }
    } else if (equationNumber == 2) {
        while (true) {...}
    } else System.out.println("Incorrect equation number.");
}
```

### 2. Метод хорд:

```

DataReceiver receiver = new DataReceiver();
double x_next = 0;
double tmp;
double[] coefficients = receiver.receiveCoefficients();
double a = coefficients[0];
double b = coefficients[1];
double e = coefficients[2];
int equationNumber = receiver.receiveEquation();
if (equationNumber == 1) {
    do {
        tmp = x_next;
        x_next = b - receiver.solveFirstEquation(b) * (a - b) / (receiver.solveFirstEquation(a) - receiver.solveFirstEquation(b));
        a = b;
        b = tmp;
    } while (Math.abs(x_next - b) > e);
}
else {
    do {
        tmp = x_next;
        x_next = b - receiver.solveSecondEquation(b) * (a - b) / (receiver.solveSecondEquation(a) - receiver.solveSecondEquation(b));
        a = b;
        b = tmp;
    } while (Math.abs(x_next - b) > e);
}
if (receiver.solveFirstEquation(x_next) == 0) {
    System.out.println("x = " + x_next + ". Is exact value.");
} else System.out.println("x = " + x_next + ". Is approximate value accurate to  $\epsilon$  = " + e + ".");

```

### 3. Метод простых итераций:

```

int step = 1;
double x1;
if (equationNumber == 1) {
    do {
        x1 = receiver.solveSecondFuncFromFirstSystem(a);
        System.out.println("Iteration #" + step + ". x1 = " + x1 + " and f(x1) = " + receiver.solveFirstFuncFromFirstSystem(x1));
        step += 1;
        if (step > b) {
            System.out.println("Not convergent.");
            System.exit(status: 0);
        }
        a = x1;
    } while (Math.abs(receiver.solveFirstFuncFromFirstSystem(x1)) > e);
    System.out.println("Root is: " + x1);
} else {
    do {
        x1 = receiver.solveSecondFuncFromSecondSystem(a);
        System.out.println("Iteration #" + step + ". x1 = " + x1 + " and f(x1) = " + receiver.solveFirstFuncFromSecondSystem(x1));
        step += 1;
        if (step > b) {
            System.out.println("Not convergent.");
            System.exit(status: 0);
        }
        a = x1;
    } while (Math.abs(receiver.solveFirstFuncFromSecondSystem(x1)) > e);
    System.out.println("Root is: " + x1);
}

```

Работа программы:

```
Lab work #2 demonstrating system is started!
-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-
Task #1.1. Usage of bisection method for nonlinear equations solving:
-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-
You need to enter a segment [a;b].
Enter "a" value: 1
Enter "b" value: 10
Enter "ε" value is the accuracy with which it is necessary to find a solution to the equation: 0.001
You need to choose some equation from list for equation solver working demonstration.
Variants:
1.  $x^2 + 5x + 7 = 0$ ;
2.  $x^3 + \sqrt{x} = 0$ ;
Enter equation number: 1
x = 5.5. Is approximate value accurate to  $\epsilon = 0.001$ .
```

```
-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|
Task #1.2. Usage of chord method for nonlinear equations solving:
-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|
You need to enter a segment [a;b].
Enter "a" value: 1
Enter "b" value: 10
Enter "ε" value is the accuracy with which it is necessary to find a solution to the equation: 0.01
You need to choose some equation from list for equation solver working demonstration.
Variants:
1.  $x^2 + 5x + 7 = 0$ ;
2.  $x^3 + \text{sqrt}(x) = 0$ ;
Enter equation number: 2
x = 0.0. Is approximate value accurate to  $\varepsilon = 0.01$ .
```

Вывод: в ходе проделанной работы я реализовал два метода (деления пополам и хорд) для решения нелинейных уравнений и один метод (простых итераций) для решения систем нелинейных уравнений. Понял, что решение СНАУ не сильно отличается от решения СЛАУ. Проблема возникает при «ручном» вычислении, однако скорость вычисления компьютером и возможность быстро провести большое число операций невелирует это неудобство. Методы деления пополам и хорд выглядят очень похожи, но лично для меня первый кажется более интуитивно понятным, но за счет этого он более медленный. Метод же простых итераций я реализовал, как итерационный метод, на каждой новой итерации использующий значения, полученные на предыдущих шагах.