

**Университет ИТМО**  
**Факультет программной инженерии и компьютерной**  
**техники**

Лабораторная работа №3  
по «Вычислительной математике»

Вариант 9

Выполнил: Кривошейкин Сергей

Группа Р3214

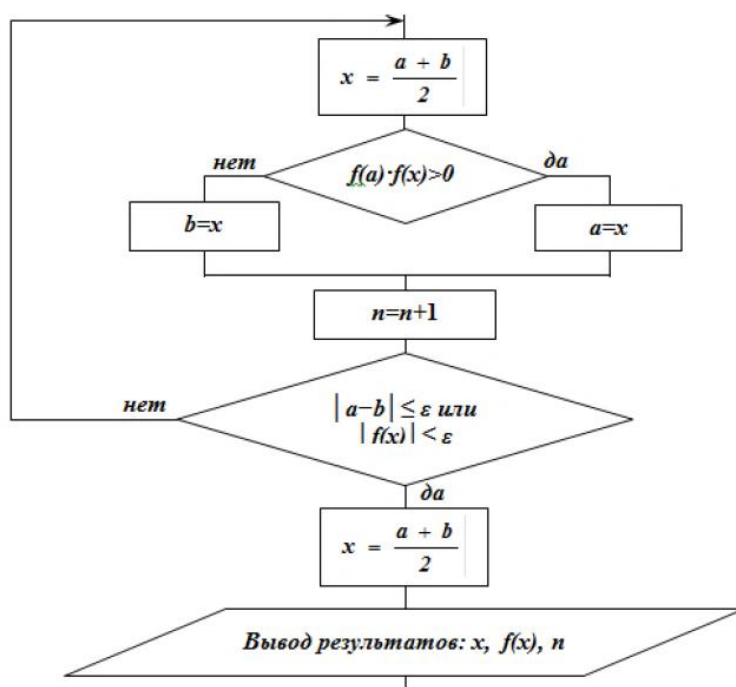
Преподаватель: Малышева Т. А.

**Санкт-Петербург**

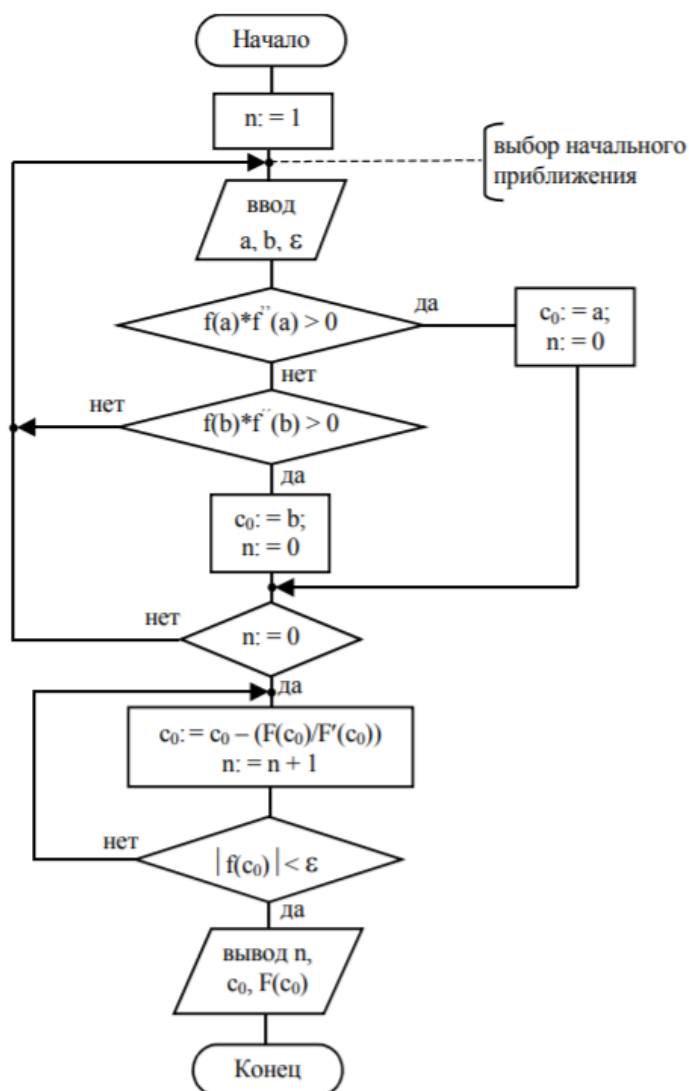
**2020**

**Цель работы:** решить нелинейное уравнение

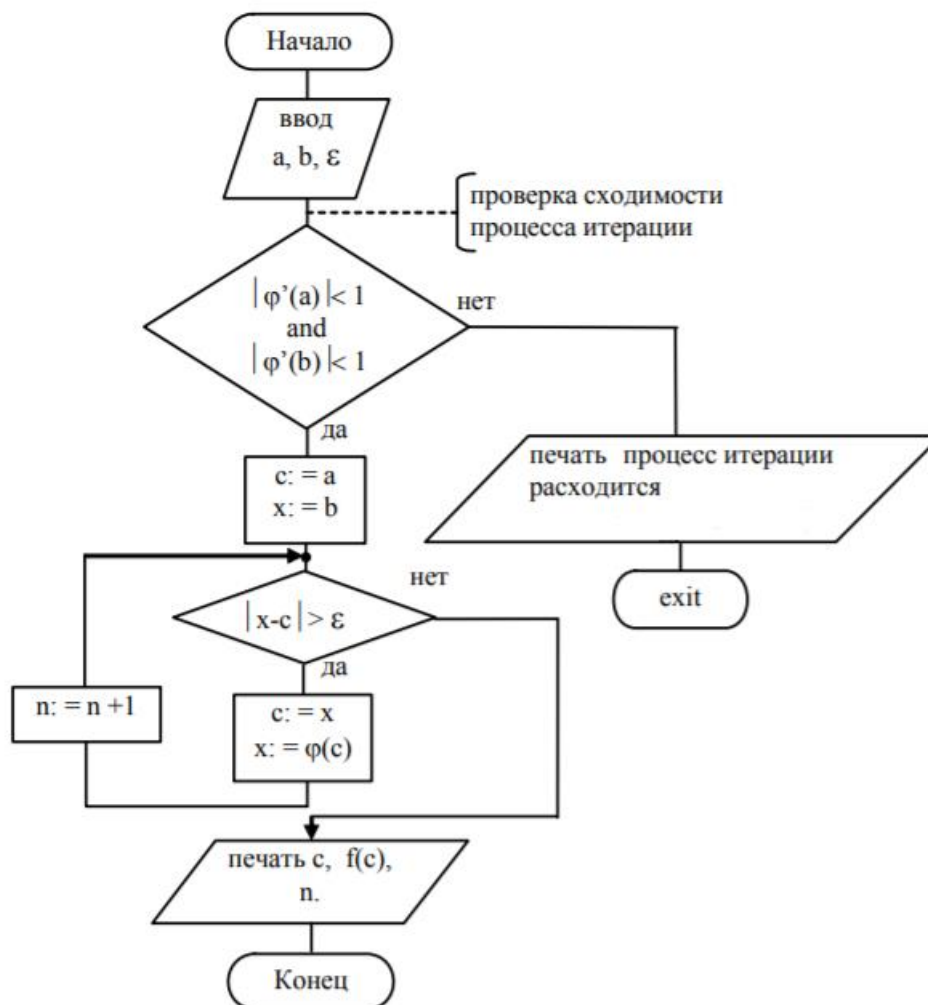
**Метод половинного деления:**



**Метод Ньютона:**



## Метод простой итерации:



### 1. Уточнение крайнего правого корня уравнения методом половинного деления

№ шага	a	b	x	f(a)	f(b)	f(x)	a-b
0	1	3	2	11.01	-38.57	-0.04	2
1	1	2	1.5	11.01	-0.04	8.245	1
2	1.5	2	1.75	8.245	-0.04	4.876	0.5
3	1.75	2	1.875	4.876	-0.04	2.622	0.25
4	1.875	2	1.937	2.622	-0.04	1.354	0.125
5	1.937	2	1.968	1.354	-0.04	0.681	0.063
6	1.968	2	1.984	0.681	-0.04	0.324	0.032
7	1.984	2	1.992	0.324	-0.04	0.143	0.016
8	1.992	2	1.996	0.143	-0.04	0.051	0.008

## 2. Уточнение крайнего левого корня уравнения методом Ньютона

№ итерации	$x_k$	$f(x_k)$	$f'(x_k)$	$x_{k+1}$	$ x_k - x_{k+1} $
0	-3,5	10,245	-35,2	-3,208	0,292
1	-3,208	1,282	-26,339	-3,159	0,049
2	-3,159	0,026	-24,943	-3,157	0,002

## 3. Уточнение центрального корня уравнения методом простой итерации

№ итерации	$x_k$	$f(x_k)$	$x_{k+1}$	$ x_k - x_{k+1} $
1	-1	-6,13	-0.435	0.565
2	-0.435	0.461	-0.477	0.042
3	-0.477	-0.046	-0.473	0.003

## Листинг программы:

### Main.java

```
public class Main {

    static Calculation calc = new Calculation();

    public static void st() {
        Scanner in = new Scanner(System.in);
        int a = 0;
        try {
            do {
                System.out.print("Меню: \n 1 - Начать вычисление\n 2 - Выйти\nВаш выбор: ");
                a = in.nextInt();
            } while ((a < 1) | (a > 2));
        } catch (Exception e) {
            System.out.println("Некорректный ввод!");
            st();
        }

        switch (a) {
            case 1:
                System.out.println("\nДана функция: -1.8x^3 -2.94x^2 + 10.37x + 5.38\n");
                st2();
                break;
            case 2:
                System.exit(0);
                break;
        }
    }

    public static void st2() {
        Scanner in = new Scanner(System.in);
        int a = 0;
        try {
            do {
                System.out.println("Выберите метод решения: \n 1 - Метод половинного деления\n 2 - Метод Ньютона\n 3 - Метод простой итерации\nВаш выбор: ");
                a = in.nextInt();
            } while ((a < 1) | (a > 3));
        }
```

```

    } catch (Exception e) {
        System.out.println("Некорректный ввод!");
        st2();
    }

    switch (a) {
        case 1:
            calc.half();
            break;
        case 2:
            calc.newton();
            break;
        case 3:
            calc.simple();
            break;
    }
}

public static void main(String[] args) {
    st();
}
}

```

### Calculation.java

```

public class Calculation {

    double a = 0;
    double b = 0;
    double E = 0;
    double x = 0;
    int n = 0;
    Graph graph = new Graph();

    public double f(double x) {
        return -1.8 * Math.pow(x, 3) - 2.94 * Math.pow(x, 2) + 10.37 * x + 5.38;
    }

    public double df(double x) {
        return -5.4 * Math.pow(x, 2) - 5.88 * x + 10.37;
    }

    public double ddf(double x) {
        return -10.8 * x - 5.88;
    }

    public double dfi(double x, double L) {
        return -5.4 * L * Math.pow(x, 2) - 5.88 * L * x + 10.37 * L + 1;
    }

    public void in() {
        Scanner in = new Scanner(System.in);
        int q = 0;
        try {
            do {
                System.out.println("\nКаким образом вы хотите ввести данные?\n 1 - Из файла \n 2 - С клавиатуры");
                q = in.nextInt();
            } while ((q < 1) | (q > 2));
        } catch (Exception e) {
            System.out.println("Некорректный ввод!");
            in();
        }
    }
}

```

```

switch (q) {
    case 1:
        System.out.println("Данные в файле должны быть разделены пробелом и
представлены в последовательности a b E");
        System.out.print("Введите имя файла: ");
        Scanner inn = new Scanner(System.in);
        String filename = inn.nextLine();
        try {
            FileInputStream fstream = new FileInputStream(filename);
            BufferedReader br = new BufferedReader(new InputStreamReader(fstream));
            String line = "";
            while ((line = br.readLine()) != null) {
                String[] numbers = line.split(" ");
                a = Double.parseDouble(numbers[0]);
                b = Double.parseDouble(numbers[1]);
                E = Double.parseDouble(numbers[2]);
            }
        } catch (FileNotFoundException e1) {
            System.out.println("Файл не найден!");
            System.exit(0);
        } catch (Exception e) {
            System.out.println("Возникли проблемы с открытием файла!");
            System.exit(0);
        }
        graph.build();
        break;
    case 2:
        Scanner innn = new Scanner(System.in);
        System.out.println("При вводе дробного числа отделяйте целую часть запятой!");
        System.out.println("Введите левую границу:");
        a = innn.nextDouble();
        System.out.println("Введите правую границу:");
        b = innn.nextDouble();
        System.out.println("Введите желаемую погрешность:");
        E = innn.nextDouble();
        graph.build();
        break;
}

if (a > b) {
    double tmp = a;
    a = b;
    b = tmp;
}

public void out(double x, int n) {
    Scanner in = new Scanner(System.in);
    int w = 0;
    try {
        do {
            System.out.println("\nКаким образом вы хотите вывести решение?\n 1 - В файл \n
2 - В консоль");
            w = in.nextInt();
        } while ((w < 1) | (w > 2));
    } catch (Exception e) {
        System.out.println("Некорректный ввод!");
        out(x, n);
    }

    switch (w) {
        case 1:
            System.out.println("Введите имя файла:");
            String filename = in.next();
            try {

```

```

        FileOutputStream fos = new FileOutputStream(filename);
        PrintStream filePrintStream = new PrintStream(fos);
        filePrintStream.println("Найденный корень уравнения = " + x + "\nЗначение
функции в корне = " + f(x) + "\nЧисло итераций = " + n);
    } catch (Exception e) {
        System.out.println("Ошибка чтения файла!");
        System.exit(0);
    }
    break;
    case 2:
        System.out.println("Найденный корень уравнения = " + x + "\nЗначение функции в
корне = " + f(x) + "\nЧисло итераций = " + n);
        break;
    }
}

public void half() {
    in();

    if (f(a) * f(b) > 0) {
        System.out.println("Корней на данном промежутке нет!");
        System.exit(0);
    } else {
        while (Math.abs(b - a) > E) {
            x = (a + b) / 2;
            n++;
            if (f(a) * f(x) < 0) {
                b = x;
            } else {
                a = x;
            }
        }
        x = (a + b) / 2;
    }
    out(x, n);
}

public void newton() {
    in();
    if (f(a) * f(b) > 0) {
        System.out.println("Корней на данном промежутке нет!");
        System.exit(0);
    }
    if (f(a) * ddf(a) > 0) {
        x = a;
    } else if (f(b) / ddf(b) > 0) {
        x = b;
    } else {
        System.out.println("Невозможно вычислить корень!");
        System.exit(0);
    }
    while (Math.abs(f(x)) >= E) {
        x = x - f(x) / df(x);
        n++;
    }
    x = x - f(x) / df(x);
    out(x, n);
}

public void simple() {
    in();

    if (f(a) * f(b) > 0) {
        System.out.println("Корней на данном промежутке нет!");
        System.exit(0);
    }
}

```

```

    }

    double L = -1 / Math.max(df(a), df(b));

    if (dfi(a, L) < 1 && dfi(b, L) < 1) {
        double c = a;
        x = b;
        while (Math.abs(x - c) > E) {
            c = x;
            x = c + L * f(c);
            n++;
        }
        c = x;
        x = c + L * f(c);
        out(x, n);
    } else {
        System.out.println("Процесс итерации расходится, ответ вычислить невозможно");
        System.exit(0);
    }
}
}

```

## Graph.java

```

public class Graph {

    public double f(double x) {
        return -1.8 * Math.pow(x, 3) - 2.94 * Math.pow(x, 2) + 10.37 * x + 5.38;
    }

    public void build() {
        String name = "-1.8x^3 -2.94x^2 + 10.37x + 5.38";

        XYSeries series = new XYSeries(name);

        for (double i = -4; i < 3; i += 0.2) {
            series.add(i, f(i));
        }

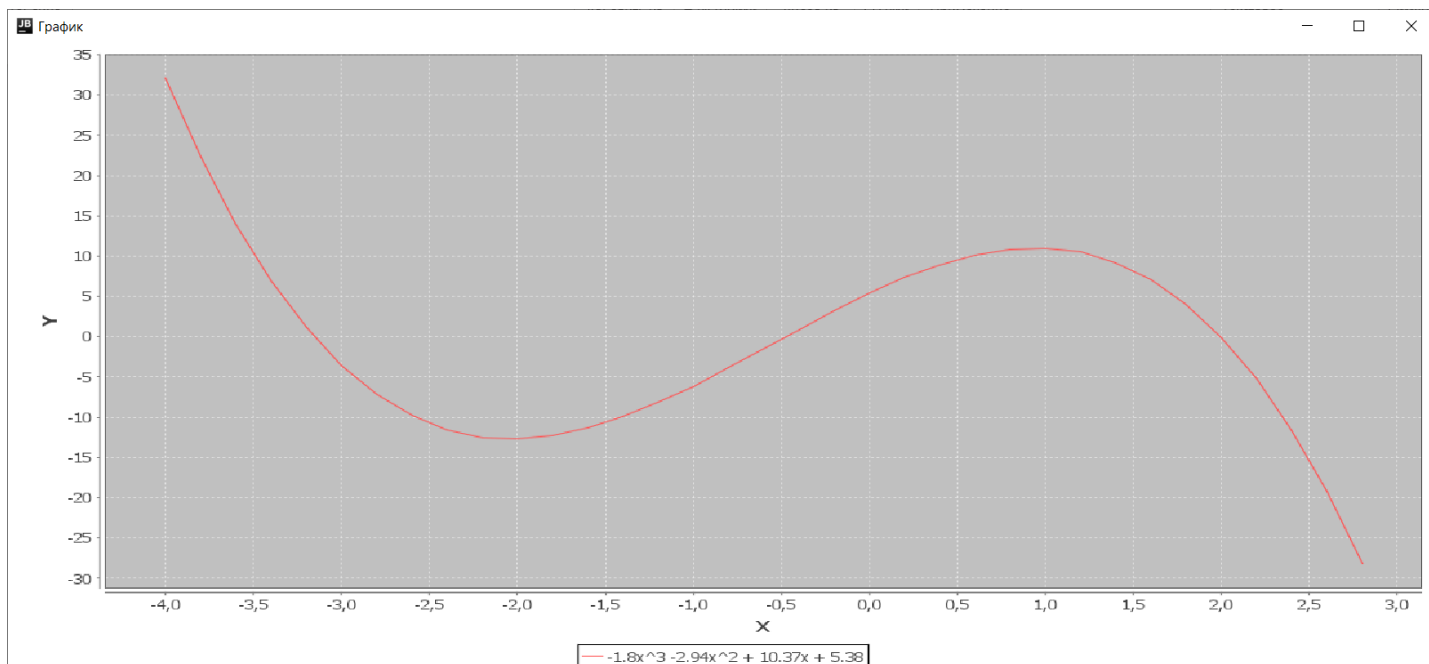
        XYDataset dataset = new XYSeriesCollection(series);

        JFreeChart chart = ChartFactory.createXYLineChart("", "X", "Y", dataset,
        PlotOrientation.VERTICAL, true, true, true);

        JFrame frame = new JFrame("График");
        frame.getContentPane().add(new ChartPanel(chart));
        frame.setSize(1300, 600);
        frame.show();
    }
}

```





## Результаты выполнения программы:

<p>Меню:</p> <ul style="list-style-type: none"> <li>1 - Начать вычисление</li> <li>2 - Выйти</li> </ul> <p>Ваш выбор: 1</p> <p>Дана функция: <math>-1.8x^3 - 2.94x^2 + 10.37x + 5.38</math></p> <p>Выберите метод решения:</p> <ul style="list-style-type: none"> <li>1 - Метод половинного деления</li> <li>2 - Метод Ньютона</li> <li>3 - Метод простой итерации</li> </ul> <p>Ваш выбор: 1</p> <p>Каким образом вы хотите ввести данные?</p> <ul style="list-style-type: none"> <li>1 - Из файла</li> <li>2 - С клавиатуры</li> </ul> <p>1</p> <p>Данные в файле должны быть разделены пробелом и представлены в последовательности a b E</p> <p>Введите имя файла: input</p> <p>Каким образом вы хотите вывести решение?</p> <ul style="list-style-type: none"> <li>1 - В файл</li> <li>2 - В консоль</li> </ul> <p>2</p> <p>Найденный корень уравнения = 1.99609375</p> <p>Значение функции в корне = 0.049595139026640034</p> <p>Число итераций = 9</p>	<p>Меню:</p> <ul style="list-style-type: none"> <li>1 - Начать вычисление</li> <li>2 - Выйти</li> </ul> <p>Ваш выбор: 1</p> <p>Дана функция: <math>-1.8x^3 - 2.94x^2 + 10.37x + 5.38</math></p> <p>Выберите метод решения:</p> <ul style="list-style-type: none"> <li>1 - Метод половинного деления</li> <li>2 - Метод Ньютона</li> <li>3 - Метод простой итерации</li> </ul> <p>Ваш выбор: 1</p> <p>Каким образом вы хотите ввести данные?</p> <ul style="list-style-type: none"> <li>1 - Из файла</li> <li>2 - С клавиатуры</li> </ul> <p>2</p> <p>При вводе дробного числа отделяйте целую часть запятой!</p> <p>Введите левую границу: -2</p> <p>Введите правую границу: -1</p> <p>Введите желаемую погрешность: 0,01</p> <p>Корней на данном промежутке нет!</p>
---	---

<p>Меню:</p> <p>1 - Начать вычисление</p> <p>2 - Выйти</p> <p>Ваш выбор: 1</p> <p>Дана функция: <math>-1.8x^3 - 2.94x^2 + 10.37x + 5.38</math></p> <p>Выберите метод решения:</p> <p>1 - Метод половинного деления</p> <p>2 - Метод Ньютона</p> <p>3 - Метод простой итерации</p> <p>Ваш выбор: 3</p> <p>Каким образом вы хотите ввести данные?</p> <p>1 - Из файла</p> <p>2 - С клавиатуры</p> <p>2</p> <p>При вводе дробного числа отделяйте целую часть запятой!</p> <p>Введите левую границу:</p> <p>-1</p> <p>Введите правую границу:</p> <p>0</p> <p>Введите желаемую погрешность:</p> <p>0,001</p> <p>Каким образом вы хотите вывести решение?</p> <p>1 - В файл</p> <p>2 - В консоль</p> <p>2</p> <p>Найденный корень уравнения = -</p> <p>0.4736478210311174</p> <p>Значение функции в корне = -</p> <p>2.7544379450183953E-5</p> <p>Число итераций = 4</p>	<p>Меню:</p> <p>1 - Начать вычисление</p> <p>2 - Выйти</p> <p>Ваш выбор: 1</p> <p>Дана функция: <math>-1.8x^3 - 2.94x^2 + 10.37x + 5.38</math></p> <p>Выберите метод решения:</p> <p>1 - Метод половинного деления</p> <p>2 - Метод Ньютона</p> <p>3 - Метод простой итерации</p> <p>Ваш выбор: 2</p> <p>Каким образом вы хотите ввести данные?</p> <p>1 - Из файла</p> <p>2 - С клавиатуры</p> <p>2</p> <p>При вводе дробного числа отделяйте целую часть запятой!</p> <p>Введите левую границу:</p> <p>-4</p> <p>Введите правую границу:</p> <p>-3</p> <p>Введите желаемую погрешность:</p> <p>0,001</p> <p>Каким образом вы хотите вывести решение?</p> <p>1 - В файл</p> <p>2 - В консоль</p> <p>2</p> <p>Найденный корень уравнения = -</p> <p>3.157946119027653</p> <p>Значение функции в корне =</p> <p>7.194245199571014E-14</p> <p>Число итераций = 4</p>
--	--

**Вывод:** в ходе выполнения данной лабораторной работы я реализовал программу на языке Java по решению нелинейного уравнения и открыл для себя новые методы решения.