МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

Федеральное государственное автономное образовательное учреждение высшего образования **«Национальный исследовательский университет ИТМО»**

ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ И КОМПЬЮТЕРНОЙ ТЕХНИКИ

ЛАБОРАТОРНАЯ РАБОТА №2

‘ВЫЧИСЛИТЕЛЬНАЯ МАТЕМАТИКА’

Вариант №24

Студент:
Хоанг Ван Куан
Группа P3266

Преподаватель:
Машина Екатерина Александровна

Санкт-Петербург, 2024

1. Цель работы

Изучить численные методы решения нелинейных уравнений и их систем, найти корни заданного нелинейного уравнения/системы нелинейных уравнений, выполнить программную реализацию методов.

2. Порядок выполнения работы

- 1 часть: Решение нелинейного уравнения
  Задание:
  1. Отделить корни заданного нелинейного уравнения графически

     $f(x) = x^3 - 2{,}92x^2 + 1{,}435x + 0{,}791$

  2. Определить интервалы изоляции корней.
  3. Уточнить корни нелинейного уравнения с точностью ε = 10^{-2}
  4. Используемые методы для уточнения каждого из 3-х корней многочлена представлены в таблице 7.
     Крайний правый корень: метод хорд
     Крайний левый корень: метод Ньютона
     Центральный корень: метод простой итерации
  5. Вычисления оформить в виде таблиц (1-5), в зависимости от заданного метода. Для всех значений в таблице удержать 3 знака после запятой.
     Для метода хорд заполнить таблицу 2.
     Для метода Ньютона заполнить таблицу 3.
     Для метода простой итерации заполнить таблицу 5. Проверить условие сходимости метода на выбранном интервале.

- 2 часть: Решение системы нелинейных уравнений
  Задание:
  1. Отделить корни заданной системы нелинейных уравнений графически

     $$\begin{cases} \sin(x-y) - xy = -1 \\ 0{,}3x^2 + y^2 = 2 \end{cases}$$

  2. Используя метод Ньютона, решить систему нелинейных уравнений с точностью до 0,01.

- 3 часть: Программная реализация
  Для нелинейных уравнений
  - Метод хорд
  - Метод секущих
  - Метод простой итерации

  Для систем нелнейных уравнений
  - Метод простой интерации

3. Рабочие формулы

Рабочая формула метода хорд:

$$x_{i+1} = x_i - \frac{x_0 - x_i}{f(a) - f(x_i)} f(x_i)$$

Рабочая формула метода Ньютона:

$$x_{i+1} = x_i - \frac{f(i)}{f'(x_i)}$$

Рабочая формула метод секущих

$$x_{i+1} = x_i - \frac{x_i - x_{i-1}}{f(x_i) - f(x_{i-1})} f(x_i)$$

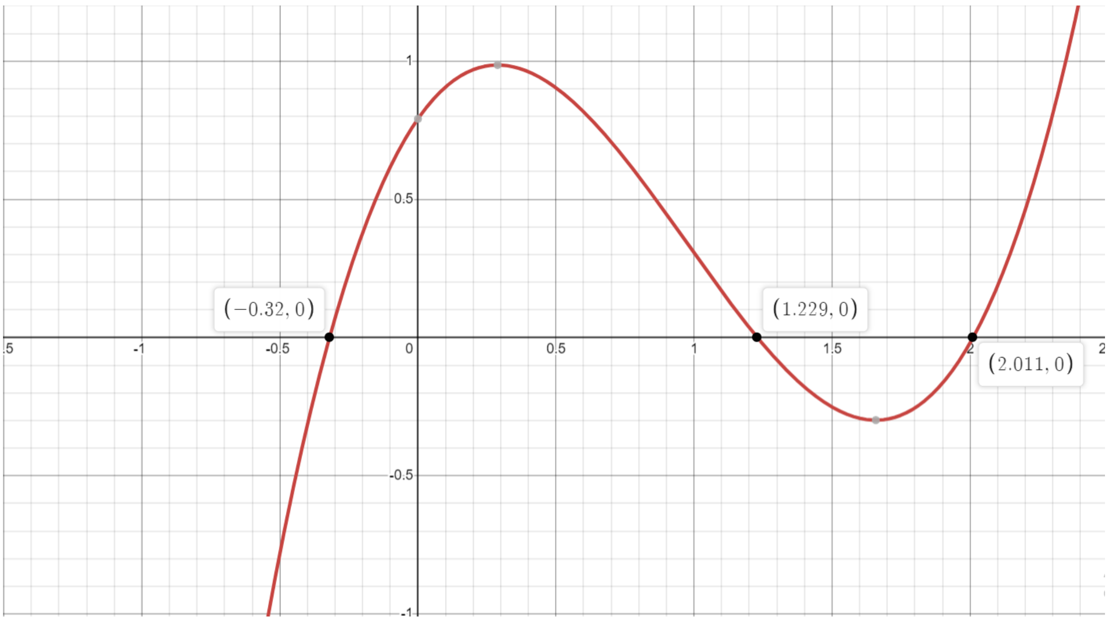
Рабочая формула метода простой интерации:

$$x_{i+1} = \varphi(x_i), \text{где } x = \varphi(x) - \text{эквивалентный вид уравнения } f(x) = 0$$

4. Вычислительная часть 1

1 - Отделить корни заданного нелинейного уравнения графически

Функция : $f(x) = x^3 - 2{,}92x^2 + 1{,}435x + 0{,}791$



2 - Определить интервалы изоляции корней

Для определения интервалов изоляции корней данного уравнения, можно воспользоваться табличным способом. Для этого нужно найти значения функции на различных интервалах и определить знак функции на каждом из них.

x	f(x)
-1	-4.564
-0.5	-0.7815
0	0.791
0.5	0.9035
1	0.306
1.5	-0.2515
2	-0.019
2.5	1.7535
3	5.816
3.5	12.9185

Знаем, что если непрерывная функция f(x) на концах отрезка [a; b] принимает значения разных знаков, т.е. f(a)·f(b)<0, то на этом отрезке содержится хотя бы один корень уравнения

Тогда мы получаем 3 интервала изоляции корней уравнения: (-0.5;0), (1;1.5) и (2;2.5)

3 - Уточнить корни нелинейного уравнения с точностью $\varepsilon = 10^{-2}$

$x_1 = -0{,}32$
$x_2 = 1.23$
$x_3 = 2.01$

4 - Используемые методы для уточнения каждого из 3-х корней

- Метод хорд

№ шага	a	b	x	$f(a)$	$f(b)$	$f(x)$	$\lvert x_{n+1} - x_n \rvert$
0	2.00000	2.50000	**2.00536**	-0.01900	1.75350	-0.00951	0.00536 < ε

$$x_i = \frac{a_i f(b_i) - b_i f(a_i)}{f(b_i) - f(a_i)}$$

$n = 0$

$\boldsymbol{x^*} \approx \mathbf{2.00536}$

- Метод Ньютона

$$x_{i+1} = x_i - \frac{f(i)}{f'(x_i)}$$

№ итерации	x_n	$f(x_n)$	$f'(x_n)$	x_{n+1}	$\lvert x_{n+1} - x_n \rvert$
0	-0.50000	-0.78150	5.10500	-0.34691	0.15309
1	-0.346910	-0.10000	3.82203	-0.32075	0.02616
2	-0.32075	-0.00269	3.61684	**-0.32001**	0.00074 < ε

$n = 2$

$\boldsymbol{x^*} \approx \mathbf{-0.32001}$

- Метод простой итерации

Преобразуем уравнение к виду

$$x = \varphi(x) = x + \frac{200}{281}(x^3 - 2{,}92x^2 + 1{,}435x + 0{,}791)$$

$a_0 = 1, b_0 = 1.5$

$$\varphi'(x) = 1 + \frac{200}{281}(3x^2 - 5.84x + 1.435) \rightarrow |\varphi'(x)| < 1 \; \forall \, x \in [1,1.5]$$

Условие сходимости выполняется

№ итерации	x_i	x_{i+1}	$\varphi(x_{i+1})$	$f(x_{i+1})$	$\lvert x_{n+1} - x_n \rvert$
0	1.00000	1.21779	1.22785	0.01413	0.21779
1	1.21779	1.22785	1.22917	0.00185	0.01006
2	1.22785	**1.22917**	1.22935	0.00025	0.00132 < ε

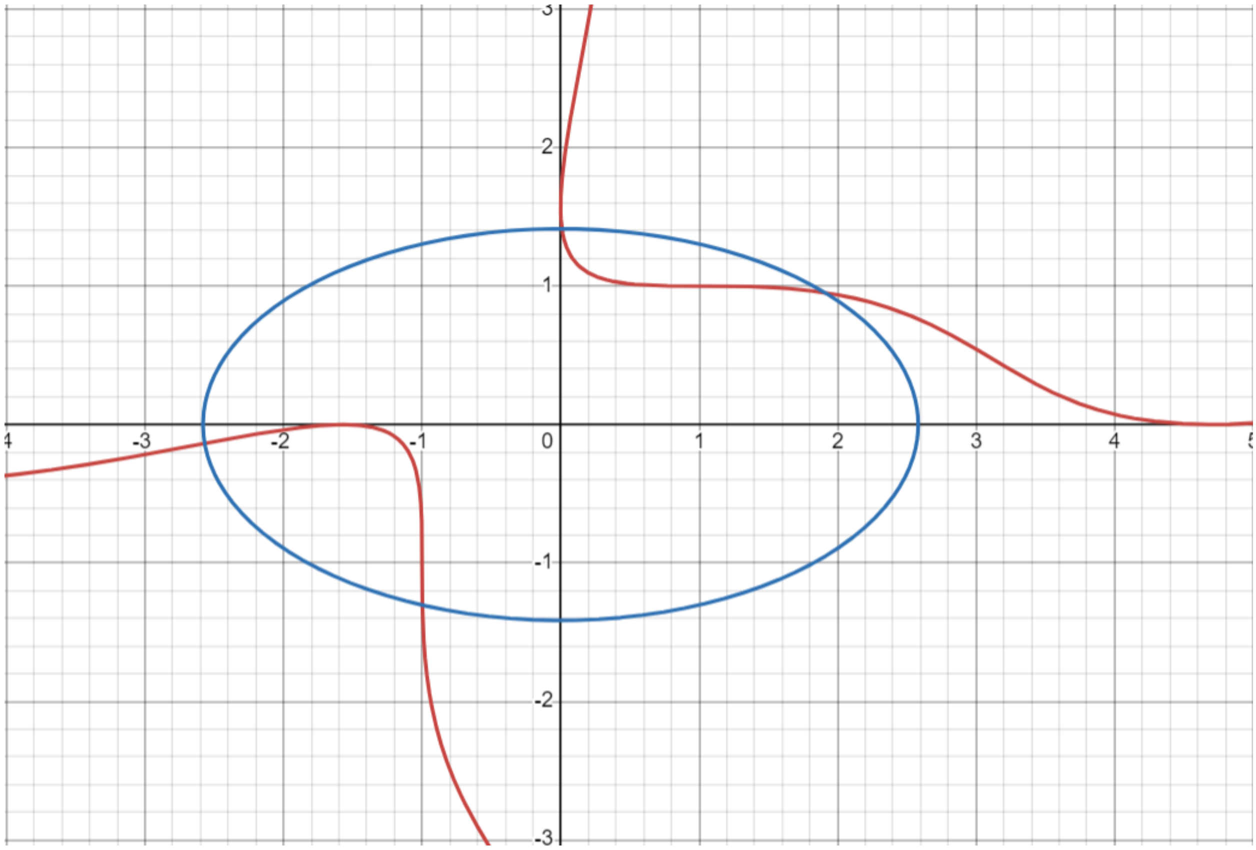
$n = 2$

$\boldsymbol{x^* \approx 1.22917}$

5. Вычислительная часть 2

1 - Отделить корни заданной системы нелинейных уравнений графически

$$\begin{cases} sin(x-y) - xy = -1 \\ 0.3x^2 + y^2 = 2 \end{cases}$$



2 – Решить систему нелинейных уравнений с точностью ε = 10^{-2} по методу Ньютона

$$\begin{cases} sin(x-y) - xy = -1 \\ 0.3x^2 + y^2 = 2 \end{cases} \rightarrow \begin{cases} sin(x-y) - xy + 1 = 0 \\ 0.3x^2 + y^2 - 2 = 0 \end{cases}$$

Построим матрицу Якоби

$$\begin{vmatrix} \frac{\partial f(x,y)}{\partial x} & \frac{\partial f(x,y)}{\partial y} \\ \frac{\partial g(x,y)}{\partial x} & \frac{\partial g(x,y)}{\partial x} \end{vmatrix} \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} = - \begin{pmatrix} f(x,y) \\ g(x,y) \end{pmatrix}$$

$$\rightarrow \begin{vmatrix} \cos(x-y) - y & -\cos(x-y) - x \\ 0.6x & 2y \end{vmatrix} \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} = \begin{pmatrix} -sin(x-y) + xy - 1 \\ -0.3x^2 - y^2 + 2 \end{pmatrix}$$

$$\begin{cases} (\cos(x-y) - y)\Delta x - (\cos(x-y) + x)\Delta y = -\sin(x-y) + xy - 1 \\ 0.6x\Delta x + 2y\Delta y = -0.3x^2 - y^2 + 2 \end{cases}$$

Выбираем $x_0 = -3, y_0 = -1$

$$\rightarrow \begin{cases} 0.584\Delta x + 3.416\Delta y = 2.909 \\ -1.8\Delta x - 2\Delta y = -1.7 \end{cases} \rightarrow \begin{cases} \Delta x = -0.002 \\ \Delta y = -0.852 \end{cases}$$

$x_1 = x_0 + \Delta x = -3.002, \quad y_0 = y_0 + \Delta y = -1.852$

Аналогично, получается таблица

№ интерации	x_i	y_i	Δx_i	Δy_i	x_{i+1}	y_{i+1}
0	-3	-1	-0.0022	0.852	-3.0022	-0.1480
1	-3.0022	-0.1480	0.4008	0.0133	-2.6014	-0.1347
2	-2.6014	-0.1347	0.0312	-0.0014	-2.5702	-0.1361
3	**-2.5702**	**-0.1361**	0.0002 < ε	-0.00005 < ε	-	-

$n = 2$

$\boldsymbol{x^* \approx -2.5702}$

$\boldsymbol{y^* \approx -0.1361}$

6. Листинг программы

Метод хорд

```
private void hordMethod() {
    ans = new Answers();
    double x = a - ((b-a) * f.apply(a) )/(f.apply(b) - f.apply(a));
    if(f.apply(a) * f.apply(x) < 0) b = x;
    if(f.apply(x) * f.apply(b) < 0) a = x;

    while(true) {
        double x1 = a - ((b - a) * f.apply(a)) / (f.apply(b) - f.apply(a));
        if (Math.abs(x1 - x) <= epsilon || Math.abs(a - b) <= epsilon ||
Math.abs(f.apply(x1)) <= epsilon) {
            ans.solution = x1;
            ans.value = f.apply(x1);
```

```
                ans.iteration = Math.abs(x1 - x);
                break;
            }
            x = x1;
            if (f.apply(a) * f.apply(x) < 0) b = x;
            if (f.apply(x) * f.apply(b) < 0) a = x;
        }
}
```

Метод секущих

```
private void secanMethod() {
    ans = new Answers();
    double x = a;
    if(f.apply(a) * f.d(2, a) > 0) x = a;
    if(f.apply(b) * f.d(2, b) > 0) x = b;
    double x1 = x - f.apply(x)/f.d(1,x);
    while(true){
        if(Math.abs(x1 - x) <= epsilon || Math.abs(f.apply(x1)) <= epsilon){
            ans.solution = x1;
            ans.value = f.apply(x1);
            ans.iteration = Math.abs(x1 - x);
            break;
        }
        double temp = x1;
        x1 -= (x1 - x)*f.apply(x1)/(f.apply(x1)-f.apply(x));
        x = temp;
    }
}
```

Метод простой интервал

```
private void simpleIterationMethod() {
    ans = new Answers();
    Phi phi = new Phi(a,b,f);
    if(Math.max(Math.abs(f.d(1,a)), Math.abs(f.d(1,b))) < 1)
System.out.println("Условие сходимости ВЫПОЛНЯЕТСЯ");
    else System.out.println("Условие сходимости НЕ ВЫПОЛНЯЕТСЯ");
    while(true){
        double x1 = phi.apply(x0);
        if(Math.abs(x1 - x0) <= epsilon && Math.abs(f.apply(x1)) <= epsilon){
            ans.solution = x1;
            ans.value = f.apply(x1);
            ans.iteration = Math.abs(x1 - x0);
            break;
        }
        x0 = x1;
    }
}
```

Метод простой интервал для решения систем нелинейных уравнений

```
private Result simpleIterationMethod(double x, double y, double epsilon){
    Result res = new Result();
    double x0 = x, y0 = y;
    if(Math.max((Math.abs(f.dx(x,y,1)) +
Math.abs(f.dy(x,y,1))),(Math.abs(f.dx(x,y,2)) + Math.abs(f.dy(x,y,2)))) >= 1)
{
        System.out.println(f.dx(x,y,1));
        System.out.println("условие сходимости итерационного процесса не
выполнено");
    }
    while(true){
        res.num++;
        x = f.g_x(x0,y0);
```

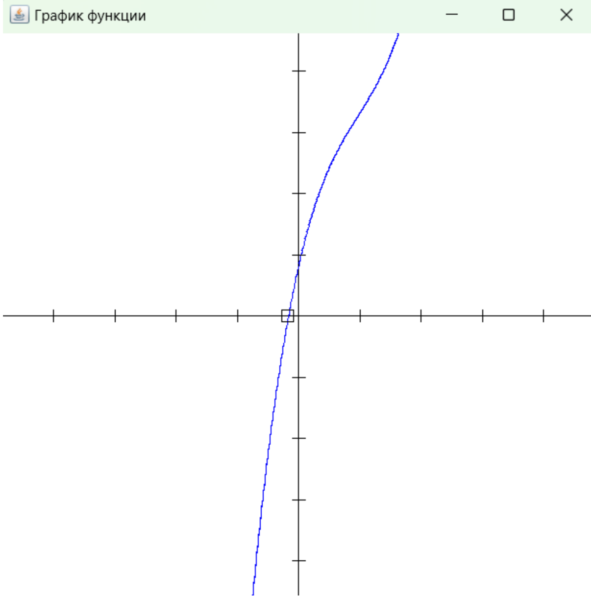
```
        y = f.g_y(x0,y0);
        if(Math.max(Math.abs(x - x0), Math.abs(y-y0)) <= epsilon){
            res.solutionX = x;
            res.solutionY = y;
            res.value1 = f.f1(x,y);
            res.value2 = f.f2(x,y);
            res.itr1 = x - x0;
            res.itr2 = y - y0;
            break;
        }
        x0 = x;
        y0 = y;
    }
    return res;
}
```

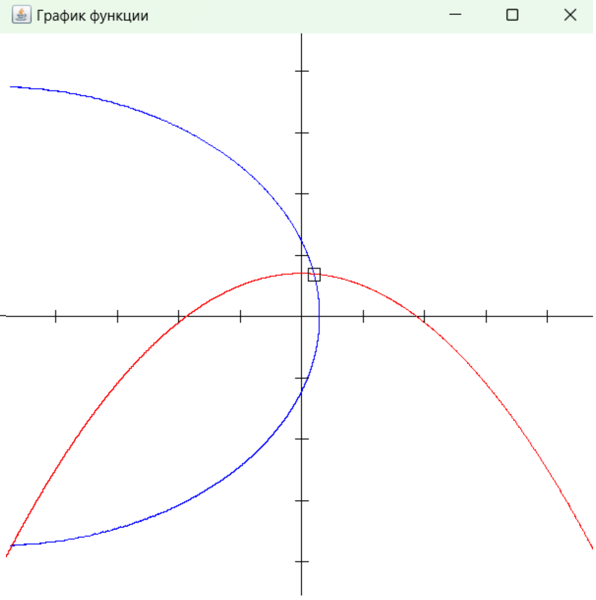
7. Результаты выполнения программы

```
ЧАСТЬ 1: НЕЛИНЕЙНОЕ УПРАВЛЕНИЕ
Взять исходные данные из файла (+) или ввести с клавиатуры (-)?
Режим ввода:
+
Вывести таблицу трассировки? (+ / -)
-
Корень уравнения: -0.16116659019524338
Значение функции в корне: -0.0038061122801095193
Число итераций: 0.00990662095053546
```

```
ЧАСТЬ 2: СИСТЕМ НЕЛИНЕЙНЫХ УРАВНЕНИЙ
Выберите функцию
1) x = 0.3 - 0.1x² - 0.2y²
   y = 0.7 - 0.2x² - 0.1xy

Выберите функцию
2) sin(y + 0.5) - x = 1
   y + cos(x - 2) = 0
Выберите функцию из списка: 1
Введите приближение x: 1
Введите приближение y: 1
Введите точность: 0.01
Вывести таблицу трассировки? (+ / -)
-
x = 0.2033827465513719 | y = 0.6773270394906847
f1: 7.264156041070224E-4
f2: 6.243888298468336E-4
Количества итераций, за которое было найдено решение: 5
Вектора погрешностей: (-0.0038773572466440642 ; -0.002090278459011219)
```

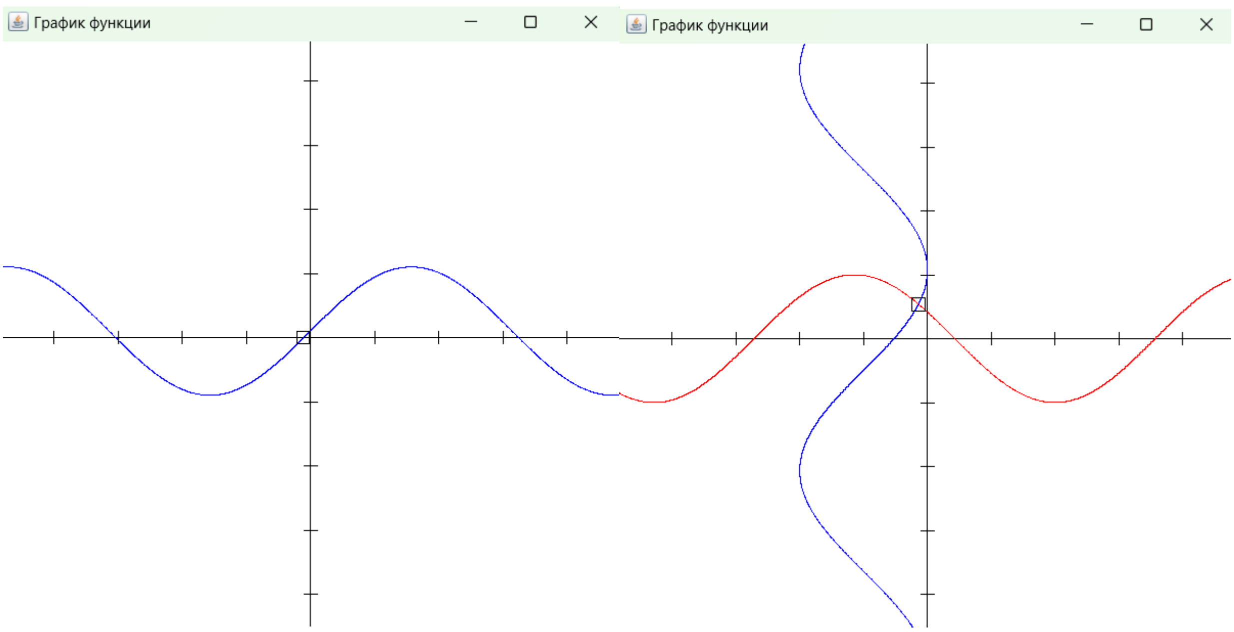





```
ЧАСТЬ 1: НЕЛИНЕЙНОЕ УПРАВЛЕНИЕ
Взять исходные данные из файла (+) или ввести с клавиатуры (-)?
Режим ввода:
-
Выберите функцию
 1) x³ - 2.92x² + 4.435x + 0.791
 2) x³ - x + 4
 3) sin(x) + 0.1
Выберите функцию из списка
3
Выберите метод решения.
 1 - Метод хорд
 2 - Метод секущих
 3 - Метод простой итерации
3
Выберите границы интервала.
Границы интервала: -5 5
Выберите начальное приближение.
Начальное приближение: 0
Выберите погрешность вычисления.
Погрешность вычисления: 0.01
Вывести таблицу трассировки? (+ / -)
-
Условие сходимости ВЫПОЛНЯЕТСЯ
Корень уравнения: -0.10106227904644849
Значение функции в корне: -8.90332196100338E-4
Число итераций: 0.0012517002434130342
```

```
ЧАСТЬ 2: СИСТЕМ НЕЛИНЕЙНЫХ УРАВНЕНИЙ
Выберите функцию
1) x = 0.3 - 0.1x² - 0.2y²
   y = 0.7 - 0.2x² - 0.1xy

Выберите функцию
2) sin(y + 0.5) - x = 1
   y + cos(x - 2) = 0
Выберите функцию из списка: 2
Введите приближение x: 1
Введите приближение y: 1
Введите точность: 0.01
Вывести таблицу трассировки? (+ / -)
-
x = -0.13374243368042982 | y = 0.5396507701372202
f1: -0.00403017889366275
f2: 0.005970776936611077
Количества итераций, за которое было найдено решение: 12
Вектора погрешностей: (0.007076130333839692 ; -0.008011293009814402)
```



8. Выводы

В ходе выполнения лабораторной работы были изучены численные методы решения нелинейных уравнений и систем нелинейных уравнений с использованием Java. Была успешно реализована программа, предусматривающая выбор уравнений, методов решения, ввод исходных данных, проверку корректности данных и сходимости методов, а также вывод результатов на экран или в файл. В результате работы были найдены корни заданных уравнений и систем с использованием различных численных методов, а также были построены графики функций для полного представления исследуемых интервалов.