Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение

Высшего образования

Факультет Программной Инженерии и Компьютерной Техники

Лабораторная работа №2 по Вычислительной Математике

Численное решение нелинейных уравнений и систем

Вариант №5

Группа: P3214

Выполнил:

Минкова Алина Андреевна

Проверил:

Малышева Татьяна Алексеевна

Г. Санкт-Петербург

2024

Оглавление

Цель работы

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

Порядок выполнения работы

1. Вычислительная реализация задачи:

Решение нелинейного уравнения:

1. Отделить корни заданного нелинейного уравнения графически.
2. Определить интервалы изоляции корней.
3. Уточнить корни нелинейного уравнения с точностью $\varepsilon=10^{-2}$
4. Вычисления оформить в виде таблиц в зависимости от заданного метода. Для всех значений в таблице удержать 3 знака после запятой. Для метода простой итерации проверить условие сходимости метода на выбранном интервале.
5. Заполненные таблицы отобразить в отчете.

Решение системы нелинейных уравнений:

1. Отделить корни заданной системы нелинейных уравнений графически.
2. Используя указанный метод, решить систему нелинейных уравнений с точностью до 0,01. Для метода простой итерации проверить условие сходимости метода.
3. Подробные вычисления привести в отчете.

2. Программная реализация задачи:

Для нелинейных уравнений:
1. Все численные методы должны быть реализованы в виде отдельных подпрограмм/методов/классов.
2. Пользователь выбирает уравнение, корень/корни которого требуется вычислить (3-5 функций, в том числе и трансцендентные), из тех, которые предлагает программа.
3. Предусмотреть ввод исходных данных (границы интервала/начальное приближение к корню и погрешность вычисления) из файла или с клавиатуры по выбору конечного пользователя.
4. Выполнить верификацию исходных данных. Необходимо анализировать наличие корня на введенном интервале. Если на интервале несколько корней или они отсутствуют – выдавать соответствующее сообщение. Программа должна реагировать на некорректные введенные данные.
5. Для методов, требующих начальное приближение к корню (методы Ньютона, секущих, хорд с фиксированным концом, простой итерации), выбор начального приближения (a или b) вычислять в программе.
6. Для метода простой итерации проверять достаточное условие сходимости метода на введенном интервале.
7. Предусмотреть вывод результатов (найденный корень уравнения, значение функции в корне, число итераций) в файл или на экран по выбору конечного пользователя.
8. Организовать вывод графика функции, график должен полностью отображать весь исследуемый интервал (с запасом).

Для систем нелинейных уравнений:
1. Пользователь выбирает предлагаемые программой системы двух нелинейных уравнений (2-3 системы).
2. Организовать вывод графика функций.
3. Начальные приближения ввести с клавиатуры.
4. Для метода простой итерации проверить достаточное условие сходимости.
5. Организовать вывод вектора неизвестных: x_1, x_2.
6. Организовать вывод количества итераций, за которое было найдено решение.
7. Организовать вывод вектора погрешностей: $|x_i^{(k)} - x_i^{(k-1)}|$.
8. Проверить правильность решения системы нелинейных уравнений.

Рабочие формулы используемых методов

Для нелинейных уравнений:

Метод половинного деления

Идея метода: начальный интервал изоляции корня делим пополам, получаем начальное приближение к корню:

$$x_0 = \frac{a_0+b_0}{2}$$

Вычисляем $f(x_0)$. В качестве нового интервала выбираем ту половину отрезка, на концах которого функция имеет разные знаки: $[a_0,x_0]$ либо $[b_0,x_0]$. Другую половину отрезка $[a_0, b_0]$, на которой функция $f(x)$ знак не меняет, отбрасываем. Новый интервал вновь делим пополам, получаем очередное приближение к корню: $x_1=(a_1+b_1)/2$. и т.д.

Рабочая формула метода: $$x_i = \frac{a_i + b_i}{2}$$

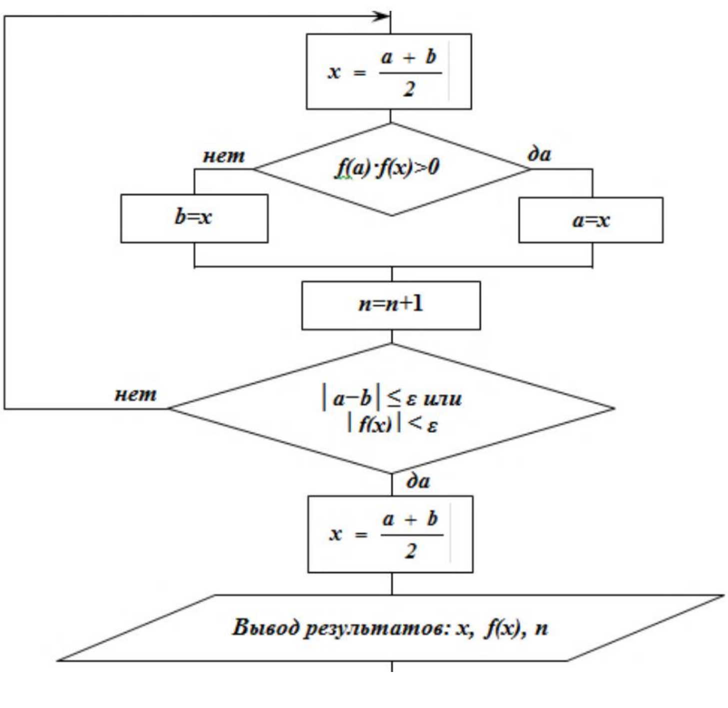
Приближенное значение корня: $x^* = \frac{a_n+b_n}{2}$ или $x^* = a_n$ или $x^* = b_n$

Критерии окончания итерационного процесса

Сходимость итерационного процесса фиксируется следующими способами:

1. Сходимость по аргументу: $|x_n - x_{n-1}| \le \varepsilon$
2. Сходимость по функции: $|f(x_n)| \le \varepsilon$

Для метода половинного деления можно рассматривать еще один критерий окончания итерационного процесса:
$|a_n - b_n| \le \varepsilon$



Метод хорд

Идея метода: функция $y = f(x)$ на отрезке [a, b] заменяется хордой и в качестве приближенного значения корня принимается точка пересечения хорды с осью абсцисс.

Уравнение хорды, проходящей через точки $A\big(a, f(a)\big)$ и $B\big(b, f(b)\big)$:

$$\frac{y-f(a)}{f(b)-f(a)} = \frac{x-a}{b-a}$$

Точка пересечения хорды с осью абсцисс ($y = 0$): $\quad x = a - \frac{b-a}{f(b)-f(a)} f(a)$

Алгоритм метода:

0 шаг: Находим интервал изоляции корня $[a_0, b_0]$

1 шаг: Вычисляем x_0: $\quad x_0 = a_0 - \frac{b_0-a_0}{f(b_0)-f(a_0)} f(a_0)$

2 шаг: Вычисляем $f(x_0)$.

3 шаг: В качестве нового интервала выбираем ту половину отрезка, на концах которого функция имеет разные знаки: $[a_0,x_0]$ либо $[b_0,x_0]$.

4 шаг: Вычисляем x_1 и т.д (повторяем 1-3 шаги).

Рабочая формула метода:

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

Критерии окончания итерационного процесса: $|x_n - x_{n-1}| \le \varepsilon$ или $|a_n - b_n| \le \varepsilon$ или $|f(x_n)| \le \varepsilon$

Приближенное значение корня: $x^* = x_n$

Семейство *хорд* может строиться:

а) при фиксированном левом конце *хорд*, тогда x_0=b (рис. 1а)

Рабочая формула метода:

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

б) при фиксированном правом конце *хорд*, тогда x_0=a (рис. 1б)

Рабочая формула метода:

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

В этом случае **НЕ** надо определять на каждой итерации новые значения a, b

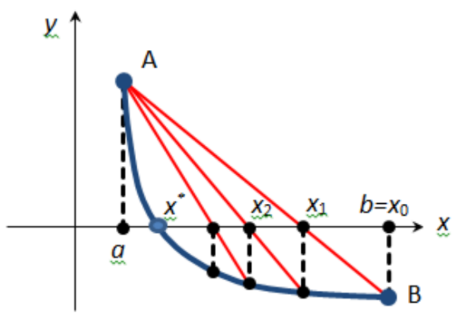


Рис. 1а

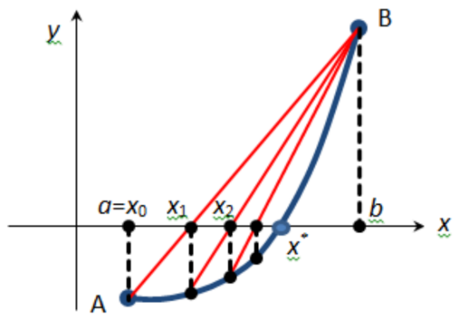


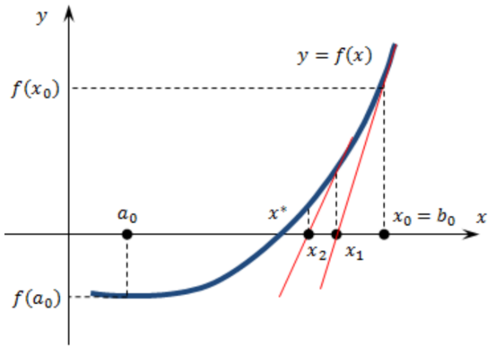
Рис. 1б

Метод Ньютона (касательных)

Идея метода: функция $y = f(x)$ на отрезке [a, b] заменяется касательной и в качестве приближенного значения корня принимается точка пересечения касательной с осью абсцисс.

Пусть $x_0 \in [a, b]$ - начальное приближение. Запишем уравнение касательной к графику функции $y = f(x)$в этой точке:

$$y = f(x_0) + f'(x_0)(x - x_0)$$



Найдем пересечение касательной с осью х:

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$$

Рабочая формула метода:

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

Критерий окончания итерационного процесса:

$|x_n - x_{n-1}| \le \varepsilon$ или $\left|\frac{f(x_n)}{f'(x_n)}\right| \le \varepsilon$ или $|f(x_n)| \le \varepsilon$

Приближенное значение корня: $x^* = x_n$

Условия сходимости метода Ньютона

Сходимость метода Ньютона зависит от того, насколько близко к корню выбрано начальное приближение. Тогда скорость сходимости велика.

Метода Ньютона эффективен, если выполняются условия сходимости:

- производные $f'(x)$ и $f''(x)$ сохраняют знак на отрезке $[a; b]$,
- производная $f'(x) \neq 0$.

Выбор начального приближения $x_0 \in [a; b]$:

Метод обеспечивает быструю *сходимость*, если выполняется условие:

$$f(x_0) \cdot f''(x_0) > 0$$

(тот конец интервала, для которого знаки функции и второй производной совпадают)

$$x_0 = \begin{cases} a_0, & \text{если } f(a_0) \cdot f''(a_0) > 0 \\ b_0, & \text{если } f(b_0) \cdot f''(b_0) > 0 \end{cases}$$

Метод секущих

Упростим метод Ньютона, заменив $f'(x)$ разностным приближением:

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

Рабочая формула метода:

$$\boldsymbol{x_{i+1} = x_i - \frac{x_i - x_{i-1}}{f(x_i) - f(x_{i-1})} f(x_i)} \qquad i = 1,2 \ldots$$

Метод секущих является *<u>двухшаговым</u>*, т.е. новое приближение x_{i+1} определяется двумя предыдущими итерациями x_i и x_{i-1}.

Выбор x_0 определяется как и в методе Ньютона, x_1- выбирается рядом с начальным самостоятельно.

Критерий окончания итерационного процесса:

$$|x_n - x_{n-1}| \le \varepsilon \quad \text{или} \quad \boldsymbol{|f(x_n)| \le \varepsilon}$$

Приближенное значение корня: $x^* = x_n$

Метод простой итерации

Уравнение $f(x) = 0$ приведем к эквивалентному виду: $x = \varphi(x)$, выразив x из исходного уравнения.

Зная начальное приближение: $x_0 \in [a, b]$, найдем очередные приближения:

$x_1 = \varphi(x_0) \to x_2 = \varphi(x_1) \ldots$

Рабочая формула метода: $\boldsymbol{x_{i+1} = \varphi(x_i)}$

Условия сходимости метода простой итерации определяются следующей теоремой.

Теорема. Если на отрезке локализации $[a, b]$ функция $\varphi(x)$ определена, непрерывна и дифференцируема и удовлетворяет неравенству:

$|\varphi'(x)| < q$, где $0 \le q < 1$, то независимо от выбора начального приближения $x_0 \in [a, b]$ итерационная последовательность $\{x_n\}$ метода будет сходится к корню уравнения.

<u>Достаточное</u> условие сходимости метода:

$|\varphi'(x)| \le q < 1$, где q – некоторая константа (коэффициент Липшица или коэффициент сжатия)

$q = \max\limits_{[a,b]} |\varphi'(x)|$

При $q \approx 0$ - скорость сходимости высокая,

При $q \approx 1$ - скорость сходимости низкая,

При $q > 1$ - нет сходимости.

Чем меньше q, тем выше скорость сходимости.

Критерий окончания итерационного процесса:

$|x_n - x_{n-1}| \le \varepsilon$ (при $0 < q \le 0{,}5$)

$|x_n - x_{n-1}| < \frac{1-q}{q}\varepsilon$ (при $0{,}5 < q < 1$)

Можно ограничиться: $|x_n - x_{n-1}| \le \varepsilon$

Для систем нелинейных уравнений:

РЕШЕНИЕ СИСТЕМЫ НЕЛИНЕЙНЫХ УРАВНЕНИЙ.
МЕТОД НЬЮТОНА.

К основе метода лежит использование разложения функций $F_i(x_1, x_2, \dots, x_n)$ в ряд Тейлора в окрестности некоторой фиксированной точки, причем члены, содержащие вторые (и более высоких порядков) производные, отбрасываются.

Пусть начальные приближения неизвестных системы (1) получены и равны соответственно $a_1, a_2, \dots, a_n$. Задача состоит в нахождении приращений (поправок) к этим значениям $\Delta x_1, \Delta x_2, \dots, \Delta x_n$, благодаря которым решение системы запишется в виде

$$x_1 = a_1 + \Delta x_1, \; x_2 = a_2 + \Delta x_2, \dots, x_n = a_n + \Delta x_n \qquad (2)$$

Проведем разложение левых частей уравнений (1) с учетом (2) в ряд Тейлора, ограничиваясь лишь линейными членами относительно приращений:

$$\begin{cases} F_1(x_1, x_2, \dots, x_n) \approx F_1(a_1, a_2, \dots, a_n) + \dfrac{\partial F_1}{\partial x_1}\Delta x_1 + \cdots \dfrac{\partial F_1}{\partial x_n}\Delta x_n \\ F_2(x_1, x_2, \dots, x_n) \approx F_2(a_1, a_2, \dots, a_n) + \dfrac{\partial F_2}{\partial x_1}\Delta x_1 + \cdots \dfrac{\partial F_2}{\partial x_n}\Delta x_n \\ \dots\dots\dots\dots\dots\dots\dots. \\ F_n(x_1, x_2, \dots, x_n) \approx F_n(a_1, a_2, \dots, a_n) + \dfrac{\partial F_n}{\partial x_1}\Delta x_1 + \cdots \dfrac{\partial F_n}{\partial x_n}\Delta x_n \end{cases}$$

Поскольку в соответствии с (1) левые части этих выражений должны обращаться в нуль, то приравняем к нулю и правые части. Получим следующую систему линейных алгебраических уравнений относительно приращений:

$$\begin{cases} \dfrac{\partial F_1}{\partial x_1}\Delta x_1 + \dfrac{\partial F_1}{\partial x_2}\Delta x_2 + \cdots + \dfrac{\partial F_1}{\partial x_n}\Delta x_n = -F_1 \\ \dfrac{\partial F_2}{\partial x_1}\Delta x_1 + \dfrac{\partial F_2}{\partial x_2}\Delta x_2 + \cdots \dfrac{\partial F_2}{\partial x_n}\Delta x_n = -F_2 \\ \dots\dots\dots\dots\dots\dots\dots. \\ \dfrac{\partial F_n}{\partial x_1}\Delta x_1 + \dfrac{\partial F_n}{\partial x_2}\Delta x_2 + \cdots \dfrac{\partial F_n}{\partial x_n}\Delta x_n = -F_n \end{cases} \qquad (3)$$

Значения $F_1, F_2, \dots, F_n$ и их производные вычисляются при $x_1 = a_1, \, x_2 = a_2, \dots, x_n = a_n$.
Определителем системы (3) является ***якобиан***:

$$J = \begin{vmatrix} \dfrac{\partial F_1}{\partial x_1} & \dots\dots & \dfrac{\partial F_1}{\partial x_n} \\ \dfrac{\partial F_2}{\partial x_1} & \dots\dots & \dfrac{\partial F_2}{\partial x_n} \\ \dots\dots & \dots\dots & \dots\dots \\ \dfrac{\partial F_n}{\partial x_1} & \dots\dots & \dfrac{\partial F_n}{\partial x_n} \end{vmatrix}$$

Итерационный процесс решения систем нелинейных уравнений методом Ньютона состоит в определении приращений $\Delta x_1, \Delta x_2, \dots, \Delta x_n$ к значениям неизвестных на каждой итерации.

Критерий окончания итерационного процесса: $max|\Delta x_i \leq \varepsilon|$.

В методе Ньютона:

1. Важен удачный выбор начального приближения для обеспечения хорошей сходимости.
2. Сходимость ухудшается с увеличением числа уравнений системы.

РЕШЕНИЕ СИСТЕМЫ НЕЛИНЕЙНЫХ УРАВНЕНИЙ. МЕТОД ПРОСТОЙ ИТЕРАЦИИ.

Приведем систему уравнений к эквивалентному виду:

$$\begin{cases} F_1(x_1, x_2, \dots, x_n) = 0 \\ F_2(x_1, x_2, \dots, x_n) = 0 \\ \dots\dots\dots\dots\dots\dots\dots. \\ F_n(x_1, x_2, \dots, x_n) = 0 \end{cases} \qquad \begin{cases} x_1 = \varphi_1(x_1, x_2, \dots, x_n) \\ x_2 = \varphi_2(x_1, x_2, \dots, x_n) \\ \dots\dots\dots\dots\dots\dots\dots. \\ x_n = \varphi_n(x_1, x_2, \dots, x_n) \end{cases}$$

Или, в векторной форме: $\boldsymbol{X} = \boldsymbol{\varphi}(\boldsymbol{X})$ $\quad \boldsymbol{\varphi}(\boldsymbol{X}) = \begin{pmatrix} \varphi_1(\boldsymbol{X}) \\ \varphi_2(\boldsymbol{X}) \\ \dots\dots \\ \varphi_n(\boldsymbol{X}) \end{pmatrix}$

Если выбрано начальное приближение: $\boldsymbol{X}^{(0)} = x_1^{(0)}, x_2^{(0)}, \dots, x_n^{(0)}$,

получим первые приближения к корням:

$$\begin{cases} x_1^{(1)} = \varphi_1(x_1^0, x_2^0, \dots, x_n^0) \\ x_2^{(1)} = \varphi_2(x_1^0, x_2^0, \dots, x_n^0) \\ \dots\dots\dots\dots\dots\dots\dots. \\ x_n^{(1)} = \varphi_n(x_1^0, x_2^0, \dots, x_n^0) \end{cases}$$

Последующие приближения находятся по формулам:

$$\begin{cases} x_1^{(k+1)} = \varphi_1\left(x_1^k, x_2^k, \dots, x_n^k\right) \\ x_2^{(k+1)} = \varphi_2\left(x_1^k, x_2^k, \dots, x_n^k\right) \\ \dots\dots\dots\dots\dots\dots\dots. \\ x_n^{(k+1)} = \varphi_n\left(x_1^k, x_2^k, \dots, x_n^k\right) \end{cases} \qquad k = 0, 1, 2, \dots$$

Сходятся ли эти последовательности?

Пусть задача отделения корней уже решена и определена достаточно малая область изоляции G, в которой находится подлежащий уточнению корень.

Пусть в этой окрестности функции $\varphi_i\left(x_1^k, x_2^k, \dots, x_n^k\right)$ дифференцируемы:

$$\varphi'(x) = \begin{bmatrix} \frac{\partial\varphi_1}{\partial x_1} & \frac{\partial\varphi_1}{\partial x_2} \cdots \frac{\partial\varphi_1}{\partial x_n} \\ \frac{\partial\varphi_2}{\partial x_1} & \frac{\partial\varphi_2}{\partial x_2} \cdots \frac{\partial\varphi_2}{\partial x_n} \\ & \dots\dots\dots \\ \frac{\partial\varphi_n}{\partial x_1} & \frac{\partial\varphi_n}{\partial x_2} \cdots \frac{\partial\varphi_n}{\partial x_n} \end{bmatrix}$$

Достаточное условие сходимости итерационного процесса:

$$\max_{[x\in G]} |\varphi'(x)| \le q < 1 \text{ или } \max_{[x\in G]} \max_{[i]} \sum_{j=1}^{n} \left|\frac{\partial\varphi_i(X)}{\partial x_j}\right| \le q < 1$$

$$\begin{gathered} \left|\frac{\partial\varphi_1}{\partial x_1}\right| + \left|\frac{\partial\varphi_1}{\partial x_2}\right| + \cdots + \left|\frac{\partial\varphi_1}{\partial x_n}\right| < 1 \\ \left|\frac{\partial\varphi_2}{\partial x_1}\right| + \left|\frac{\partial\varphi_2}{\partial x_2}\right| + \cdots + \left|\frac{\partial\varphi_2}{\partial x_n}\right| < 1 \\ \dots\dots\dots \\ \left|\frac{\partial\varphi_n}{\partial x_1}\right| + \left|\frac{\partial\varphi_n}{\partial x_2}\right| + \cdots + \left|\frac{\partial\varphi_n}{\partial x_n}\right| < 1 \end{gathered}$$

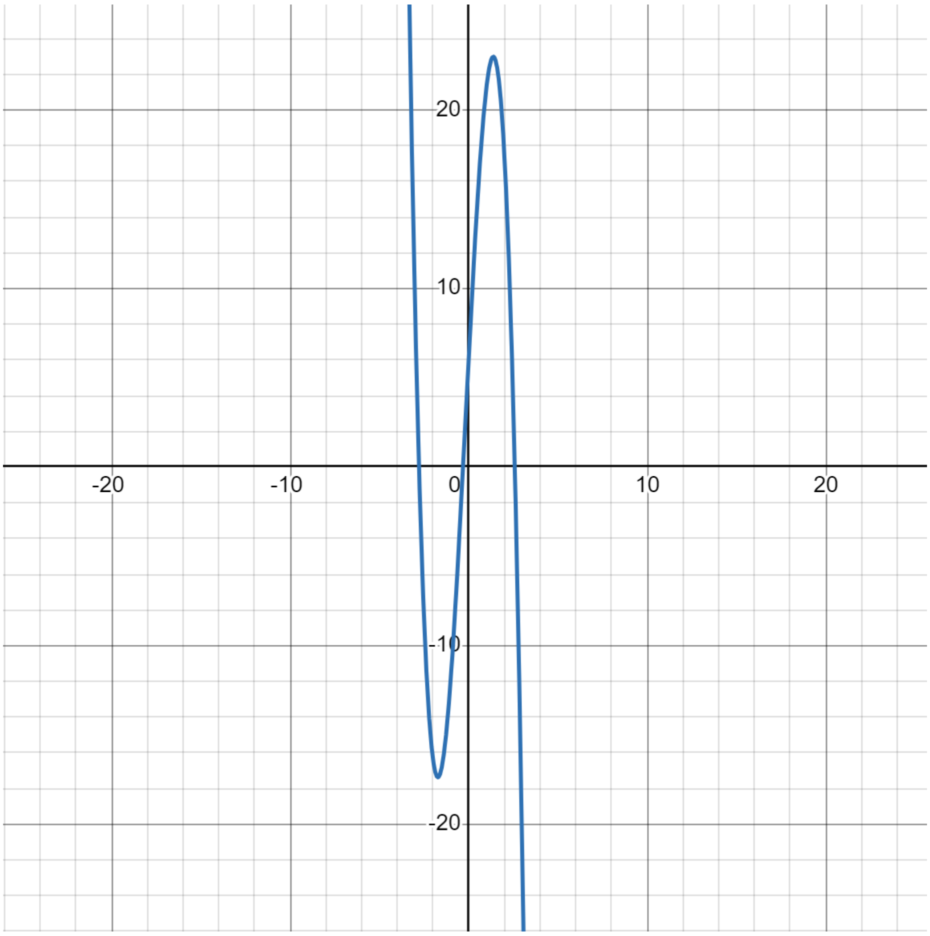
Если $\boldsymbol{X}^{(0)}$ и все последовательные приближения: $\boldsymbol{X}^{(k+1)} = \boldsymbol{\varphi}\left(\boldsymbol{X}^k\right)$, $k = 0, 1, 2 \dots$ принадлежат ограниченной замкнутой области G, тогда итерационный процесс сходится к единственному решению уравнения $\boldsymbol{X} = \boldsymbol{\varphi}(\boldsymbol{X})$

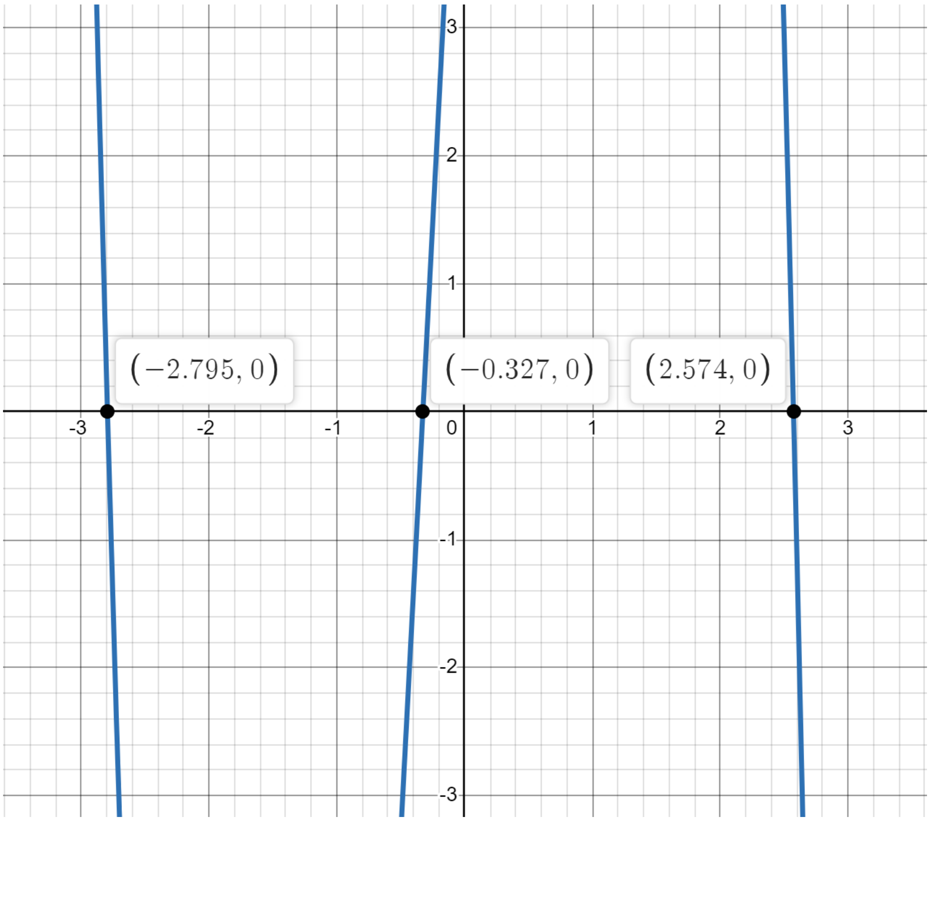
Критерий окончания итерационного процесса:

$$\max_{1<i<n} \left|x_i^{(k+1)} - x_i^k\right| \le \varepsilon$$

Графики функций на исследуемом интервале

Исследуемое уравнение: $f(x) = -2{,}7x^3 - 1{,}48x^2 + 19{,}23x + 6{,}35$





Заполненные таблицы вычислительной части 1

Интервалы изоляции корней:

- Для крайнего левого корня: $x_1 \in [-3, -2]$
- Для центрального корня: $x_2 \in [-1, 0]$
- Для крайнего правого корня: $x_3 \in [2, 3]$

Уточнение крайнего правого корня с точностью $\varepsilon = 10^{-2}$ методом хорд

№ итерации	a	b	x	f(a)	f(b)	f(x)	$\lvert x_{i+1} - x_i \rvert$
0	2,000	3,000	2,438	17,290	-22,180	5,308	0,333
1	2,438	3,000	2,547	5,308	-22,180	1,133	0,109
2	2,547	3,000	2,569	1,133	-22,180	0,223	0,022
3	2,569	3,000	2,573	0,223	-22,180	0,043	0,004

Правый корень x = 2,573

Уточнение крайнего левого корня с точностью $\varepsilon = 10^{-2}$ методом простой итерации

$f(x) = -2{,}7x^3 - 1{,}48x^2 + 19{,}23x + 6{,}35$

$f'(x) = -8{,}1x^2 - 2{,}96x + 19{,}23$

$f'(x) < 0$ на $[a, b] \to \lambda = \dfrac{1}{\max|f'(x)|}$

$f'(-3) = -44{,}79; \quad f'(-2) = -7{,}25$

$\lambda = \dfrac{1}{|-44{,}79|} = \dfrac{100}{4479}$

$x = x + \lambda f(x) \to x = x + \lambda(-2{,}7x^3 - 1{,}48x^2 + 19{,}23x + 6{,}35)$

$= -\dfrac{90}{1493}x^3 - \dfrac{140}{4479}x^2 + \dfrac{2134}{1493}x + \dfrac{635}{4479}$

$\varphi(x) = -\dfrac{90}{1493}x^3 - \dfrac{140}{4479}x^2 + \dfrac{2134}{1493}x + \dfrac{635}{4479}$

$x_0 = -3$

№ итерации	x_i	x_{i+1}	$\varphi(x_{i+1})$	$f(x_{i+1})$	$\lvert x_{i+1} - x_i \rvert$
0	-3,000	-2,799	-2,782	0,170	0,200
1	-2,799	-2,782	-2,781	-0,464	0,018
2	-2,782	-2,781	-2,781	-0,496	0,001

Левый корень x = -2,781

Уточнение центрального корня с точностью $\varepsilon = 10^{-2}$ методом секущих

$f(x) = -2{,}7x^3 - 1{,}48x^2 + 19{,}23x + 6{,}35$

$f'(x) = -8{,}1x^2 - 2{,}96x + 19{,}23$

$f''(x) = -16{,}2x - 2{,}96$

$f(-1) \cdot f''(-1) = -11{,}66 \cdot 13{,}24 = -154{,}3784 < 0$

$f(0) \cdot f''(0) = 6{,}35 \cdot (-2{,}96) = -18{,}796 < 0$

Производные $f'(x)$ и $f''(x)$ не сохраняют знак на отрезке $[-1, 0] \Rightarrow$ не выполняется условие сходимости.

Возьмем $x_{i-1} = -1$, $x_i = -0{,}5$.

№ итерации	x_{i-1}	x_i	x_{i+1}	$f(x_{i+1})$	$\lvert x_{i+1} - x_i \rvert$
0	-1,000	-0,500	-0,303	7,217	0,197
1	-0,500	-0,303	-0,438	-2,130	-0,135

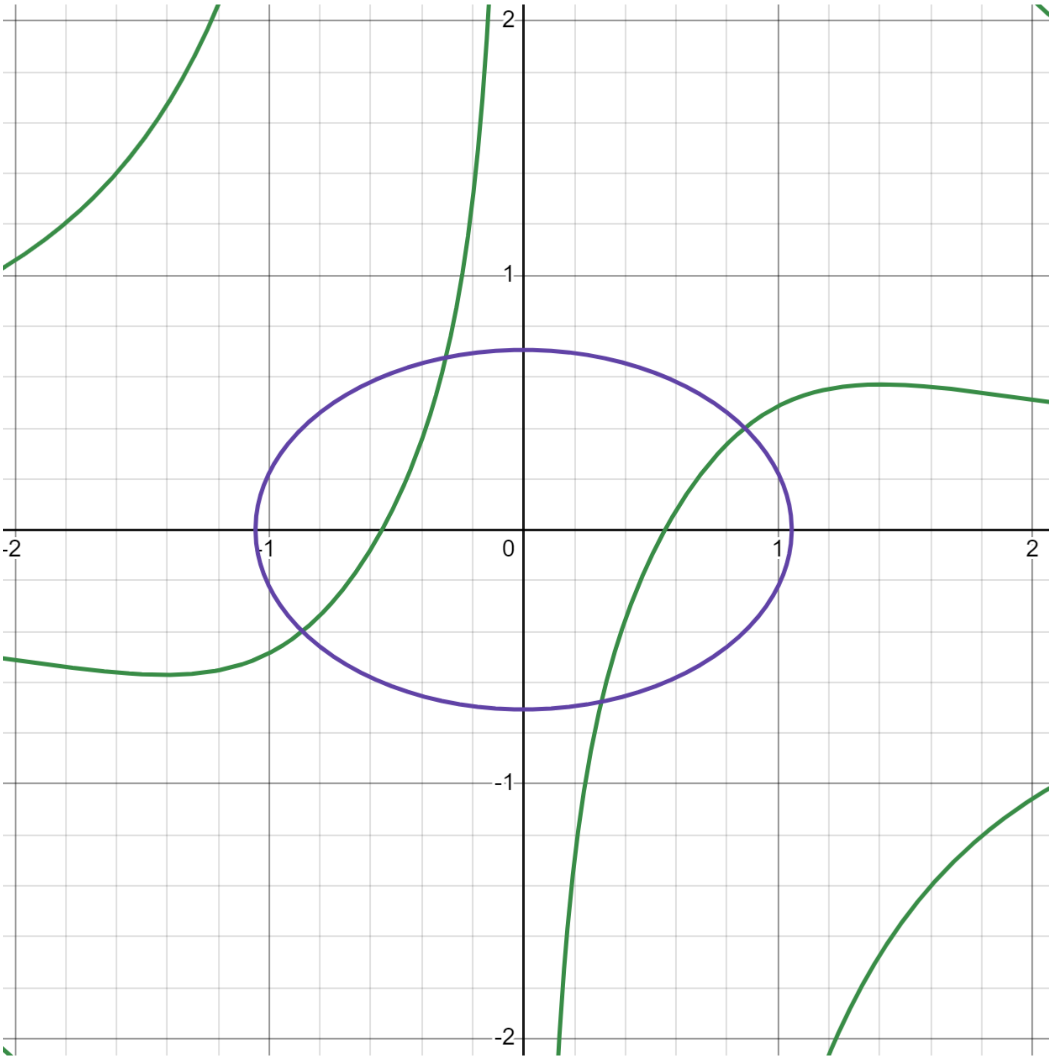
2	-0,303	-0,438	-0,407	-1,540	0,031
3	-0,438	-0,407	-0,420	-1,788	-0,013
4	-0,407	-0,420	-0,326	0,017	0,094
5	-0,420	-0,326	-0,327	-0,002	-0,001

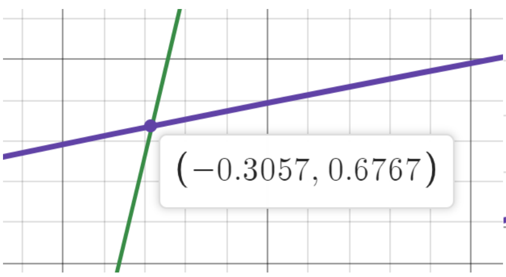
Центральный корень x = -0,327

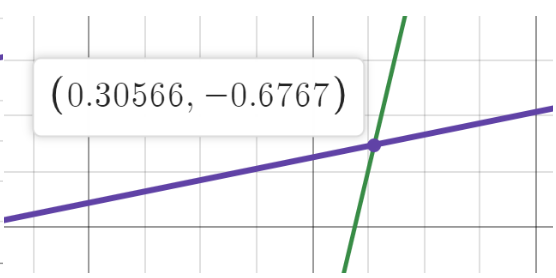
Решение системы нелинейных уравнений вычислительной части 2

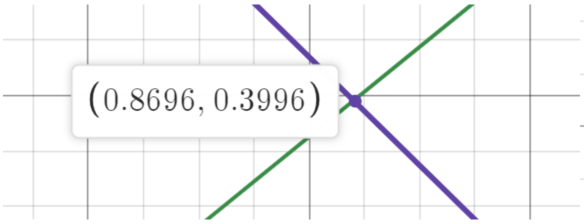
Решить следующую систему нелинейных уравнений методом Ньютона с точностью до 0,01

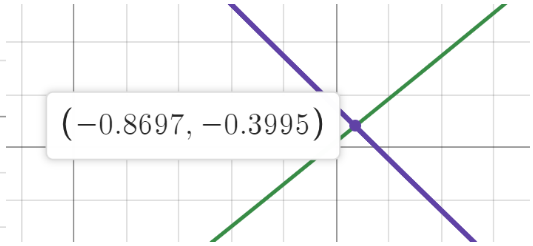
$$\begin{cases} \tan(xy + 0{,}3) = x^2 \\ 0{,}9x^2 + 2y^2 = 1 \end{cases}$$











$$\begin{cases}\tan(xy+0{,}3)=x^2\\ 0{,}9x^2+2y^2=1\end{cases} \to \begin{cases}f(x,y)=0\\ g(x,y)=0\end{cases} \to \begin{cases}\tan(xy+0{,}3)-x^2=0\\ 0{,}9x^2+2y^2-1=0\end{cases}$$

Отметим, что система имеет не более 4 различных решений.
Построим матрицу Якоби:

$$\frac{\partial f}{\partial x}=\frac{y}{\cos(xy+0{,}3)^2}-2x; \qquad \frac{\partial f}{\partial y}=\frac{x}{\cos(xy+0{,}3)^2};$$

$$\frac{\partial g}{\partial x}=1{,}8x; \qquad \frac{\partial g}{\partial y}=4y$$

Тогда будем решать следующую систему линейных уравнений:

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

$$\begin{vmatrix}\frac{y}{\cos(xy+0{,}3)^2}-2x & \frac{x}{\cos(xy+0{,}3)^2}\\ 1{,}8x & 4y\end{vmatrix}\begin{pmatrix}\triangle x\\ \triangle y\end{pmatrix}=\begin{pmatrix}x^2-\tan(xy+0{,}3)\\ 1-0{,}9x^2-2y^2\end{pmatrix}$$

или $\begin{cases}\left(\frac{y}{\cos(xy+0{,}3)^2}-2x\right)\triangle x+\left(\frac{x}{\cos(xy+0{,}3)^2}\right)\triangle y=x^2-\tan(xy+0{,}3)\\ 1{,}8x\triangle x+4y\triangle y=1-0{,}9x^2-2y^2\end{cases}$

Выбираем $x_0 = 0$, $y_0 = 1$:

1) $\begin{cases}1{,}096\triangle x+\triangle y=-0{,}309\\ \triangle x+4\triangle y=-1\end{cases} \to \begin{cases}\triangle x=-0{,}070\\ \triangle y=-0{,}233\end{cases} \to \begin{cases}x_1=-0{,}070\\ y_1=0{,}767\end{cases} \to \begin{cases}|-0{,}070|>0{,}01\\ |-0{,}233|>0{,}01\end{cases}$

2) $\begin{cases}0{,}955\triangle x-0{,}074\triangle y=-0{,}247\\ -0{,}126\triangle x+3{,}068\triangle y=-0{,}181\end{cases} \to \begin{cases}\triangle x=-0{,}264\\ \triangle y=0{,}070\end{cases} \to \begin{cases}x_2=-0{,}334\\ y_2=0{,}837\end{cases} \to \begin{cases}|-0{,}264|>0{,}01\\ |0{,}070|>0{,}01\end{cases}$

3) $\begin{cases}1{,}505\triangle x-0{,}334\triangle y=0{,}091\\ -0{,}601\triangle x+3{,}348\triangle y=-0{,}501\end{cases} \to \begin{cases}\triangle x=0{,}028\\ \triangle y=-0{,}145\end{cases} \to \begin{cases}x_3=-0{,}306\\ y_3=0{,}692\end{cases} \to \begin{cases}|0{,}028|>0{,}01\\ |-0{,}145|>0{,}01\end{cases}$

4) $\begin{cases}1{,}309\triangle x-0{,}308\triangle y=0{,}005\\ -0{,}551\triangle x+2{,}768\triangle y=-0{,}042\end{cases} \to \begin{cases}\triangle x=0{,}0003\\ \triangle y=-0{,}0015\end{cases} \to \begin{cases}x_4=-0{,}3057\\ y_4=0{,}6777\end{cases} \to \begin{cases}|0{,}0003|<0{,}01\\ |-0{,}0015|<0{,}01\end{cases}$

Решение системы: $\begin{cases}x=-0{,}3057\\ y=0{,}6777\end{cases}$

Выбираем $x_0 = 0$, $y_0 = -1$:

Аналогично предыдущим вычислениям получаем:

Решение системы: $\begin{cases}x=0{,}3057\\ y=-0{,}6777\end{cases}$

Выбираем $x_0 = 1$, $y_0 = 0$:

1) $\begin{cases}-2\triangle x+1{,}096\triangle y=0{,}691\\ 1{,}8\triangle x+\triangle y=0{,}1\end{cases} \to \begin{cases}\triangle x=-0{,}146\\ \triangle y=0{,}363\end{cases} \to \begin{cases}x_1=0{,}854\\ y_1=0{,}363\end{cases} \to \begin{cases}|-0{,}146|>0{,}01\\ |0{,}363|>0{,}01\end{cases}$

2) $\begin{cases} -1{,}168 \triangle x + 1{,}271 \triangle y = 0{,}030 \\ 1{,}537 \triangle x + 1{,}452 \triangle y = 0{,}080 \end{cases} \rightarrow \begin{cases} \triangle x = 0{,}016 \\ \triangle y = 0{,}038 \end{cases} \rightarrow \begin{cases} x_2 = 0{,}87 \\ y_2 = 0{,}401 \end{cases} \rightarrow \begin{cases} |0{,}016| > 0{,}01 \\ |0{,}038| > 0{,}01 \end{cases}$

3) $\begin{cases} -1{,}108 \triangle x + 1{,}370 \triangle y = -0{,}002 \\ 1{,}566 \triangle x + 1{,}604 \triangle y = -0{,}003 \end{cases} \rightarrow \begin{cases} \triangle x = -0{,}0002 \\ \triangle y = -0{,}0016 \end{cases} \rightarrow \begin{cases} x_3 = 0{,}8697 \\ y_3 = 0{,}3994 \end{cases} \rightarrow \begin{cases} |-0{,}0002| < 0{,}01 \\ |-0{,}0016| < 0{,}01 \end{cases}$

Решение системы: $\begin{cases} x = 0{,}8697 \\ y = 0{,}3994 \end{cases}$

Выбираем $x_0 = -1$, $y_0 = 0$:

Аналогично предыдущим вычислениям получаем:

Решение системы: $\begin{cases} x = -0{,}8697 \\ y = -0{,}3994 \end{cases}$

Листинг программы

```
def half_division(func, a0, b0, eps):
    a = a0
    b = b0
    cnt = 0
    while True:
        x = (a + b) / 2
        cnt += 1
        if func(x) == 0:
            return x
        elif func(a) * func(x) > 0:
            a = x
        else:
            b = x
        if abs(func(x)) <= eps:
            return x, func(x), cnt
```

```
def newton(func, a0, b0, eps):
    if func(a0) * diff2(func, a0) > 0:
        x = a0
    elif func(b0) * diff2(func, b0) > 0:
        x = b0
    else: x = a0
    cnt = 0
    while True:
        x = x - func(x) / diff(func, x)
        cnt += 1
        if abs(func(x)) <= eps:
            return x, func(x), cnt
```

```
def simple_iteration(func, a0, b0, eps):
    x = b0
    cnt = 0
    if diff(func, (a0 + b0) / 2) > 0:
        la = -1
    else:
        la = 1
    la *= 1 / max(abs(diff(func, a0)), abs(diff(func, b0)))
```

```
    print("fi'(a0) = ", diff_fi(la, a0, func), "\nfi'(b0) = ", diff_fi(la,
b0, func))
    flag = True
    cnt_fi = 0
    while True:
        if (abs(diff_fi(la, x, func)) > 1 or (diff_fi(la, a0, func)) > 1 or
abs(diff_fi(la, b0, func)) > 1) and flag:
            print("Условие сходимости не выполняется")
            flag = False
        x = fi(func(x), la, x)
        cnt += 1
        cnt_fi += 1
        if abs(func(x)) <= eps or cnt_fi >= 1000:
            return x, func(x), cnt
```

```
def simple_iteration_system(x1_0, x2_0, sys, eps):
    prev_x1, prev_x2 = x1_0, x2_0
    cnt = 0
    flag = True
    cnt_iter = 0
    while True:
        if (max(derivative_sys(sys, x1_0, x2_0)) > 1) and flag:
            print("Условие сходимости не выполняется")
            flag = False
        cnt += 1
        cnt_iter += 1
        x1, x2 = calculate_fx_sys(sys, prev_x1, prev_x2)
        print(x1, x2)
        del_x1 = x1 - prev_x1
        del_x2 = x2 - prev_x2
        # if abs(del_x1) and abs(del_x2) <= eps or cnt_iter >= 10000:
        #     break
        if abs(f1_1(x1, x2)) <= eps and abs(f1_2(x1, x2)) <= eps or
abs(f2_1(x1, x2)) <= eps and abs(f2_2(x1, x2)) <= eps or cnt_iter >= 100000:
            break
        prev_x1, prev_x2 = x1, x2
    return x1, x2, cnt, del_x1, del_x2
```

Результат выполнения программы

```
Выберите ввод данных:
Консоль (1)
Файл (2)
1
Выберите метод решения:
1. Метод половинного деления
2. Метод Ньютона
3. Метод простой итерации
4. Метод простой итерации для системы
2
Выберите уравнение для решения:
1. x^3 - x + 4 = 0
2. cos(x) = 0
3. -2,7x^3 -1,48x^2 + 19,23x + 6,35
2
Введите левую границу интервала: -2
Введите правую границу интервала: 0
Введите точность: 0.01
Корень уравнения: x = -1.5708040086
Значение функции в корне: f(x) = -0.0000076818
Число итераций: 2
```

```
Выберите ввод данных:
Консоль (1)
Файл (2)
1
Выберите метод решения:
1. Метод половинного деления
2. Метод Ньютона
3. Метод простой итерации
4. Метод простой итерации для системы
4
Выберите систему уравнений:
1. 0.1x1^2 + x1 + 0.2x2 - 0.3 = 0
 0.2x1^2 + x2 + 0.1x1x2 - 0.7 = 0
2. cos(x2) + x1 - 1,5 = 0
 2x2 - sin(x1 - 0,5) - 1 = 0
1
Введите начальное приближение x1_0: 1
Введите начальное приближение x2_0: 1
Введите точность: 0.01
0.2 0.6
0.1650806661517033 0.6719999999999999
0.15255685435981675 0.6780772875350537
Вектор неизвестных: x1 = 0.15255685435981675,
 x2 = 0.6780772875350537
Значения функции: f1(x1,x2) = -0.009500328751956277,
 f2(x1,x2) = -0.0069234599025952415
Количество итераций: 3
Вектор погрешностей: [-0.012523811791886552, 0.006077287535053766]
```

Вывод

В ходе выполнения лабораторной работы были реализованы численные методы решения нелинейных уравнений и систем нелинейных уравнений. Программная часть реализации методов содержит консольное приложение и графическую составляющую отображения исследуемых функций и систем.