|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

|  |  |
| --- | --- |
| ФАКУЛЬТЕТ | «Информатика и системы управления» |

|  |  |
| --- | --- |
| КАФЕДРА | «Теоретическая информатика и компьютерные технологии» |

**Отчет по лабораторной работе № 6**

**«Решение систем нелинейных уравнений методом Ньютона»**

***по курсу***

***«Численные методы»***

Выполнил:

Студент группы ИУ9-62Б

Караник А.А.

Проверила:

Домрачева А.Б.

1. *г.*

## Цель работы

Целью данной работы является реализация программы, осуществляющая решение систем нелинейных уравнений методом Ньютона.

## Постановка задачи

1. Решить систему нелинейных уравнений графически и принять полученное решение за начальное приближение
2. Решить систему методом Ньютона с точностью

Система:

## Теоретические сведения

Метод Ньютона является одним из самых распространенных методов решения системы нелинейных уравнений

или в векторном виде , где – вектор неизвестных; – вектор-функция. Выбрав начальное приближение к решению системы, следующие приближения в методе Ньютона строим по рекуррентной зависимости

где – матрица, обратная матрице Якоби .

Приближение метода Ньютона удобно искать в два этапа. Вначале решаем систему линейных уравнений с матрицей – матрицей Якоби вектор-функции :

любым из способов решения СЛАУ, например методом Гаусса. Теперь -e приближение есть сумма -го приближения и решения СЛАУ :

Для решения системы нелинейных уравнений с заданной точностью необходимо сравнить с погрешностью -го приближения

Метод Ньютона сходится, если все функции дважды непрерывно дифференцируемы по всем переменным и начальное приближение находится достаточно близко к точному решению систему. Рецепта выбора начального приближения при нет. Поэтому желательно оценить, хотя бы грубо, значение точного решения, например, решив систему графически.

## Реализация

Исходный код программы:

import numpy as np

import matplotlib.pyplot as plt

def newton\_system(F, J, x0, err=0.01):

    x = x0

    i = 0

    while True:

        delta\_x = np.linalg.solve(J(x), -F(x))

        x = x + delta\_x

        if np.linalg.norm(delta\_x) < err:

            return x, i + 1

        i += 1

def F(x):

    return np.array([

        np.cos(x[1] + 0.5) - x[0] - 2,

        np.sin(x[0]) - 2 \* x[1] - 1

    ])

def J(x):

    return np.array([

        [-1, -np.sin(x[1] + 0.5)],

        [np.cos(x[0]), -2]

    ])

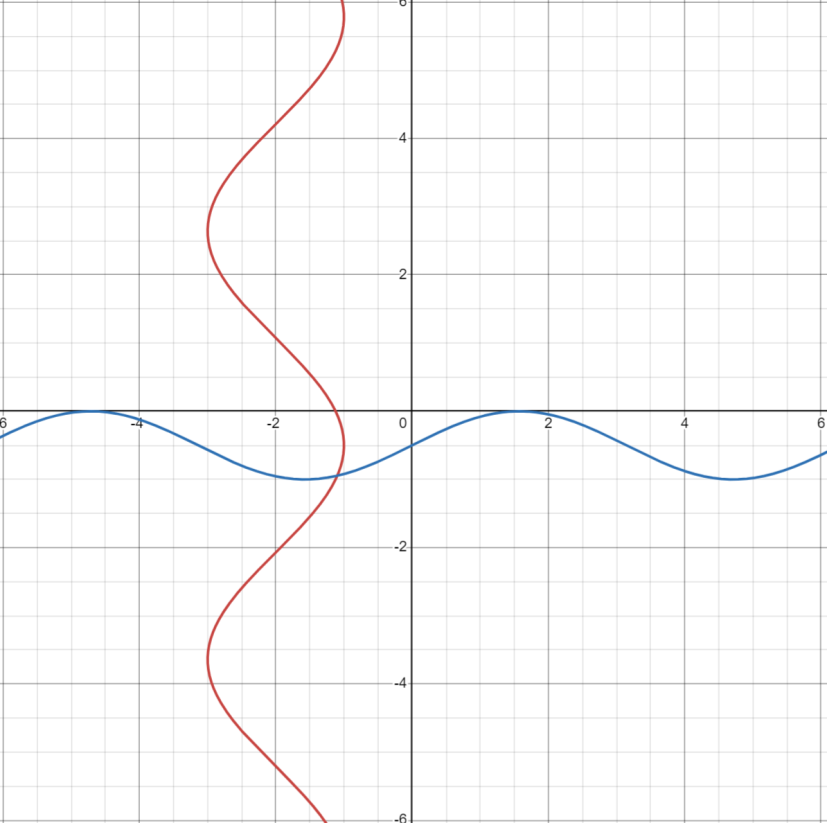
x0 = np.array([-1, -1])

solution, number = newton\_system(F, J, x0)

print("Solution:", solution)

print("Iterations:", number)

## Тестирование



Вывод программы:

Solution: [-1.09739231 -0.94501079]

Iterations: 2

## Вывод

В процессе выполнения лабораторной работы была разработана программа, осуществляющая решение систем нелинейных уравнений методом Ньютона. Этот метод является более точным по сравнению с другими. Также система была решена графически.