МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ

(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)

Институт №8 «Компьютерные науки и прикладная математика»

Кафедра 806 «Вычислительная математика и программирование»

**Лабораторные работы**

**по курсу «Численные методы»**

Выполнил: А.С. Федоров

Преподаватель: Д.Л. Ревизников

Группа: 8О-407Б Дата:

Оценка:

Подпись:

Москва, 2022

1. **Численные методы линейной алгебры**
   1. **LU-разложение матриц**

**Задача**

Реализовать алгоритм LU - разложения матриц (с выбором главного элемента) в виде программы. Используя разработанное программное обеспечение, решить систему линейных алгебраических уравнений (СЛАУ). Для матрицы СЛАУ вычислить определитель и обратную матрицу.

**Вариант 23**



Алгоритм решения

* 1. Метод прогонки

Задача

Реализовать метод прогонки в виде программы, задавая в качестве входных данных ненулевые элементы матрицы системы и вектор правых частей. Используя разработанное программное обеспечение, решить СЛАУ с трехдиагональной матрицей.

Вариант 23



Алгоритм решения

* 1. Итерационные методы решения СЛАУ

Задача

Реализовать метод простых итераций и метод Зейделя в виде программ, задавая в качестве входных данных матрицу системы, вектор правых частей и точность вычислений. Используя разработанное программное обеспечение, решить СЛАУ. Проанализировать количество итераций, необходимое для достижения заданной точности.

Вариант 23



Метод решения

* 1. Метод вращений

Задача

Реализовать метод вращений в виде программы, задавая в качестве входных данных матрицу и точность вычислений. Используя разработанное программное обеспечение, найти собственные значения и собственные векторы симметрических матриц. Проанализировать зависимость погрешности вычислений от числа итераций.

Вариант 23



Алгоритм решения

* 1. QR-алгоритм

Задача

Реализовать алгоритм QR – разложения матриц в виде программы. На его основе разработать программу, реализующую QR – алгоритм решения полной проблемы собственных значений произвольных матриц, задавая в качестве входных данных матрицу и точность вычислений. С использованием разработанного программного обеспечения найти собственные значения матрицы.

Вариант 23



Алгоритм решения

2.1 Решение нелинейных уравнений

Задача

Реализовать методы простой итерации и Ньютона решения нелинейных уравнений в виде программ, задавая в качестве входных данных точность вычислений. С использованием разработанного программного обеспечения найти положительный корень нелинейного уравнения (начальное приближение определить графически). Проанализировать зависимость погрешности вычислений от количества итераций.

Вариант 23



Метод Ньютона

Исходный код

def f(x):

    return np.log(x+2)-x\*\*4+0.5

def Newton\_method(f, df, x0, e):

    counter = 0

    x = x0

    print('Итерация '+str(counter)+': x = '+str(x))

    counter += 1

    x\_next = x-(f(x)/df(x))

    print('Итерация '+str(counter)+': x = '+str(x\_next))

    counter += 1

    while abs(x\_next - x) >= e:

        x = x\_next

        x\_next = x-(f(x)/df(x))

        print('Итерация '+str(counter)+': x = '+str(x\_next))

        counter += 1

    return x\_next

def df(x):

    return 1/(x+2)-4\*x\*\*3

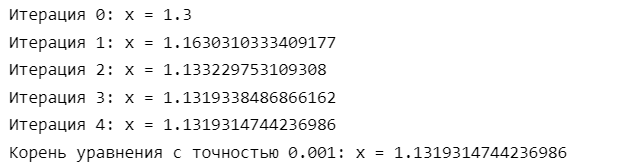
x0 = 1.3

e = 0.001

answer = Newton\_method(f, df, x0, e)

print('Корень уравнения с точностью '+str(e)+': x = '+str(answer))

Результат работы



Метод простых итераций

Исходный код

def phi(x):

    return np.power(np.log(x+2)+0.5, 1/4)

def dphi(x):

    return 1/((4\*x+8)\*np.power(np.log(x+2)+0.5, 3/4))

def Simple\_iteration\_method(phi, x0, q, e):

    counter = 0

    x = x0

    print('Итерация '+str(counter)+': x = '+str(x))

    counter += 1

    x\_next = phi(x)

    print('Итерация '+str(counter)+': x = '+str(x\_next))

    counter += 1

    while q/(1-q)\*abs(x\_next-x) > e:

        x = x\_next

        x\_next = phi(x)

        print('Итерация '+str(counter)+': x = '+str(x\_next))

        counter += 1

    return x\_next

x0 = 1.0

q = 0.0586154

e = 0.001

answer = Simple\_iteration\_method(phi, x0, q, e)

print('Корень уравнения с точностью '+str(e)+': x = '+str(answer))

Результат работы

