

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение

высшего профессионального образования

**«Дальневосточный федеральный университет»**

|  |
| --- |
| **ИНСТИТУТ МАТЕМАТИКИ И КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ**  **Департамент математического**  **и компьютерного моделирования** |

**Курсовой проект**

по дисциплине «Вычислительная математика»

на тему «LU и QR алгоритмы нахождения собственных значений несимметрической матрицы(с использованием матриц Хессенберга)»

Направление подготовки

02.03.01 «Математика и компьютерные науки»

Выполнили студенты гр. Б9119-02.03.01сцт

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_Петров С.Д.\_

(*подпись*) *(Ф.И.О.)*

\_\_\_\_\_\_\_\_\_\_\_\_\_ \_Панченко Н.К.\_

(*подпись*) *(Ф.И.О.)*

Проверил доцент, к.ф.-м.н.

Колобов А.Г.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*(подпись)*

«\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_2022г.

**г. Владивосток**

**2022**

Оглавление

[**Введение** 3](#_Toc105876809)

[**Основная часть** 4](#_Toc105876810)

[**Заключение** 9](#_Toc105876811)

[**Список использованных источников** 10](#_Toc105876812)

[**Приложения** 11](#_Toc105876813)

# **Введение**

Объектом исследования являются численными методы нахождения собственных значений матриц, а также программное обеспечение, реализующее эти методы.

Цель работы – ознакомиться с численными методами нахождения собственных значений матриц, преобразование в матрицу Хессенберга, решить предложенные типовые задачи, сформулировать выводы по полученным решениям, отметить достоинства и недостатки методов, сравнить удобство использования и эффективность работы каждой использованной программы, приобрести практические навыки и компетенции, а также опыт самостоятельной профессиональной деятельности, а именно:

* создать алгоритм решения поставленной задачи и реализовать его, протестировать программы;
* освоить теорию вычислительного эксперимента; современных компьютерных технологий;
* приобрести навыки представления итогов проделанной работы в виде отчета, оформленного в соответствии с имеющимися требованиями, с привлечением современных средств редактирования и печати.

Работа над курсовым проектом предполагает выполнение следующих задач:

* дальнейшее углубление теоретических знаний обучающихся и их систематизацию;
* получение и развитие прикладных умений и практических навыков по направлению подготовки;
* овладение методикой решения конкретных задач;
* развитие навыков самостоятельной работы;
* развитие навыков обработки полученных результатов, анализа и осмысления их с учетом имеющихся литературных данных;
* приобретение навыков оформления описаний программного продукта;
* повышение общей и профессиональной эрудиции.

Изученный студентом в ходе работы материал должен способствовать повышению его качества знаний, закреплению полученных навыков и уверенности в выборе путей будущего развития своих профессиональных способностей.

# **Основная часть**

**Постановка задачи**

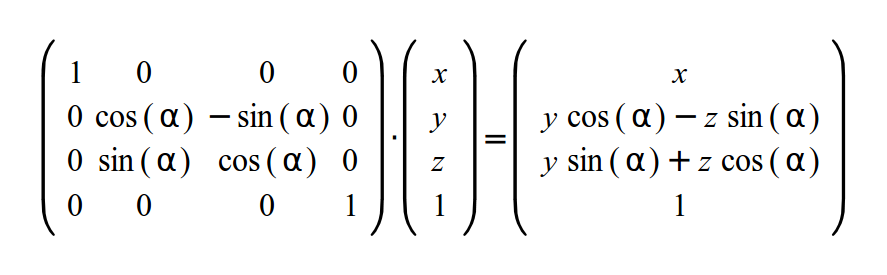
Дана несимметрическая матрица размера n × n:

Найти собственные значения этой матрицы , с помощью приведения ее к матрице Хессенберга и дальнейшего LU или QR разложения.

**Преобразование к матрице Хессенберга**

Представим матрицу как набор векторов.

Дальше создаем матрицу поворота, которая будет занулять по 1й координате.



Выберем

Отсюда

Так как и , то матрицы будут подобными, а значит их собственные значения будут равны.

**LU разложение**

При LU разложении матрица A раскладывается на произведение 2х матриц.

L – нижняя треугольная матрица, на диагонали которой лежат 1.

U – верхняя треугольная матрица.

Итерации повторяются пока выполняется условие .

Для нахождения собственных значений будет необходим итерационный процесс, где . отсюда, , а значит матрицы подобны, и их собственные значения равны. В следствии процесса получается треугольная матрица на главной диагонали которой лежат собственные значения.

**Спецификация**

– квадратная, невырожденная, все главные миноры отличны от 0.

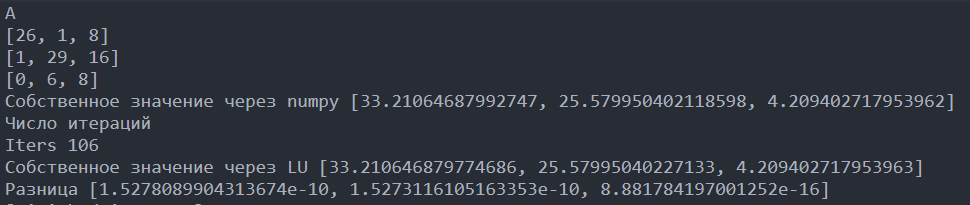
**Описание тестов**

Все вычисления будут проводиться с .

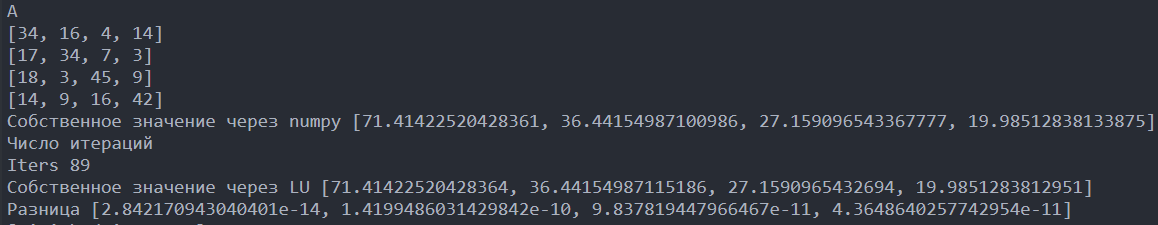
Матрицы генерируются случайным образом.

Точное решение будем находить при помощи библиотеки numpy.linalg

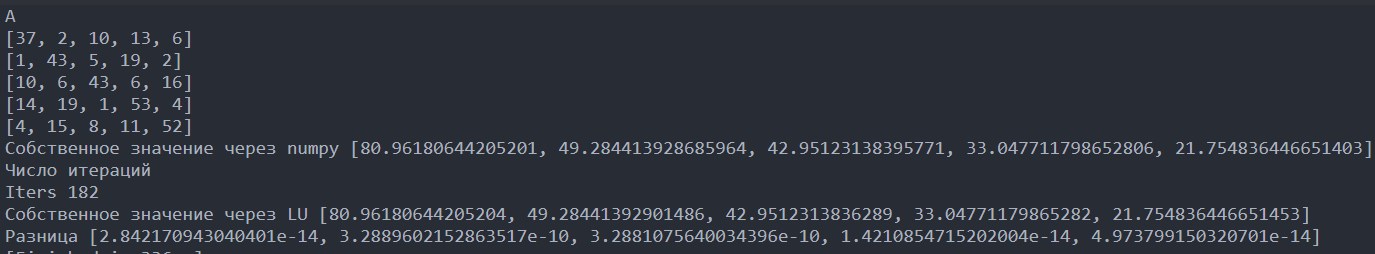
**Первый тест (n=3)**



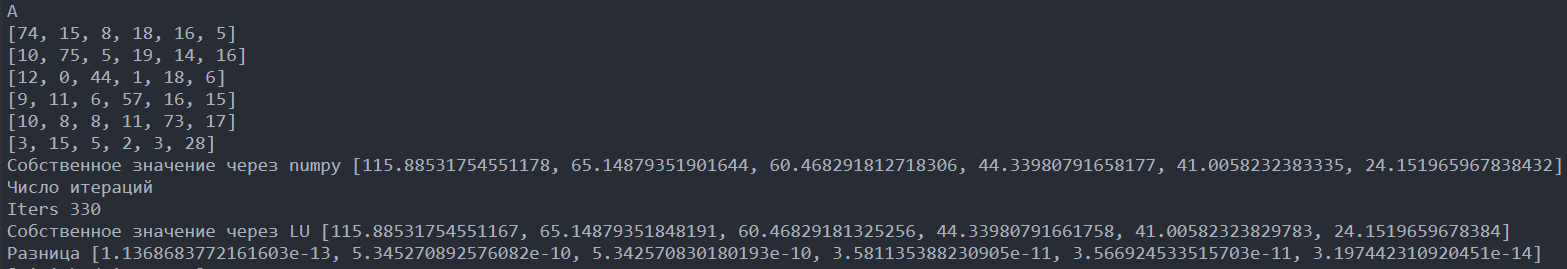
**Второй тест (n=4)**

****

**Третий тест (n=5)**



**Четвертый тест (n=6)**

****

**QR разложение**

Вычисление собственных значений матрицы *A QR* – алгоритма можно разбить на следующие этапы:

1. Уравновешивание или масштабирование матрицы:
2. Приведение матрицы к форме Хессенберга:
3. Итерационный *QR*-процесс:
4. Преобразование собственных векторов матрицы *T* в собственные векторы матрицы *A.*
5. Переход к форме Шура с заданным расположением собственных значений на главной диагонали.

Начиная с строят последовательность матриц по формулам:

Первая разложение матрицы в произведение ортогональной и правой треугольной , а вторая – перемножение полученных в результате факторизации матриц и в обратном порядке.

На основе свойства ортогональности матриц можно записать представление матрицы в виде:

*QR*-алгоритм применяется не к исходной матрице *A*, а к подобной ей правой почти треугольной матрице *B*(Хессенберга).

Итерации повторяются пока выполняется условие .

Для нахождения собственных значений будет необходим итерационный процесс, где . отсюда, , а значит матрицы подобны, и их собственные значения равны. В следствии процесса получается треугольная матрица на главной диагонали которой лежат собственные значения.

**Спецификация**

произвольная матрица размера m×n.

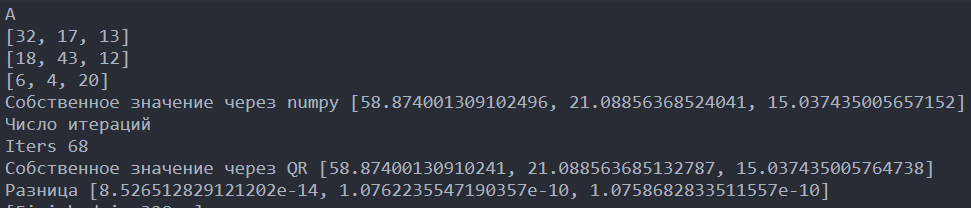
**Описание тестов**

Все вычисления так же будут проводиться с .

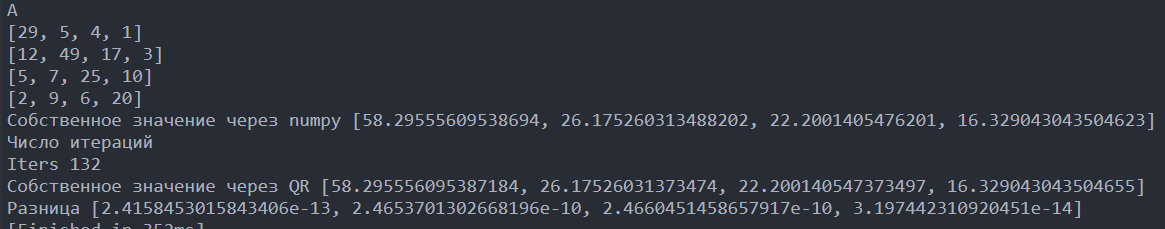
Матрицы генерируются случайным образом.

Точное решение будем находить при помощи библиотеки numpy.linalg

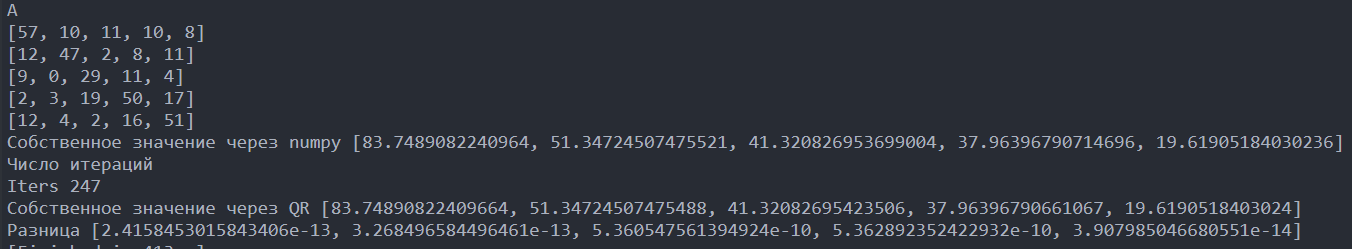
**Первый тест (n=3)**

****

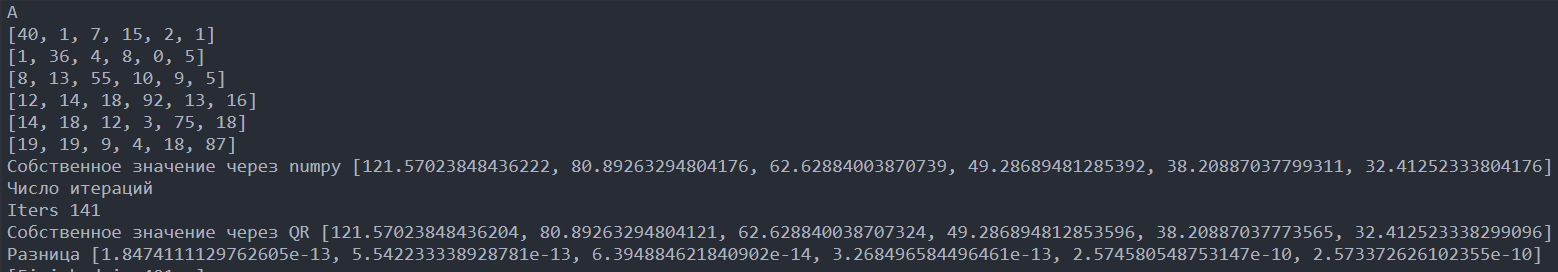
**Второй тест (n=4)**

****

**Третий тест (n=5)**

****

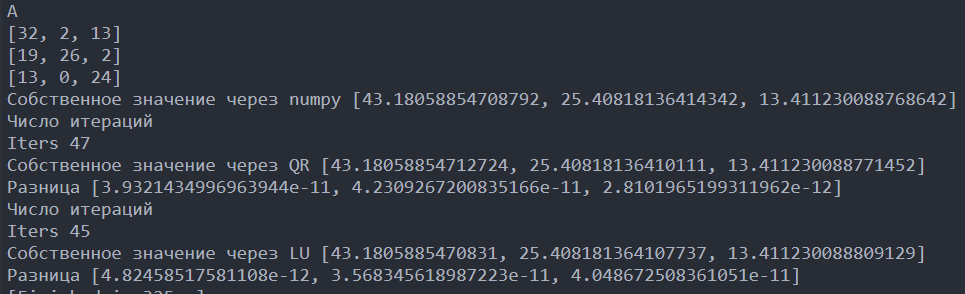
**Четвертый тест (n=6)**

****

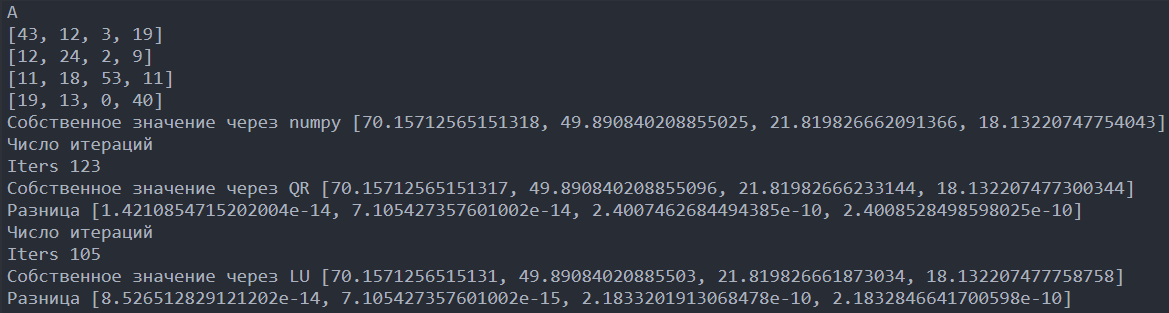
**Совместное тестирование**

Описание тестов остается прежним

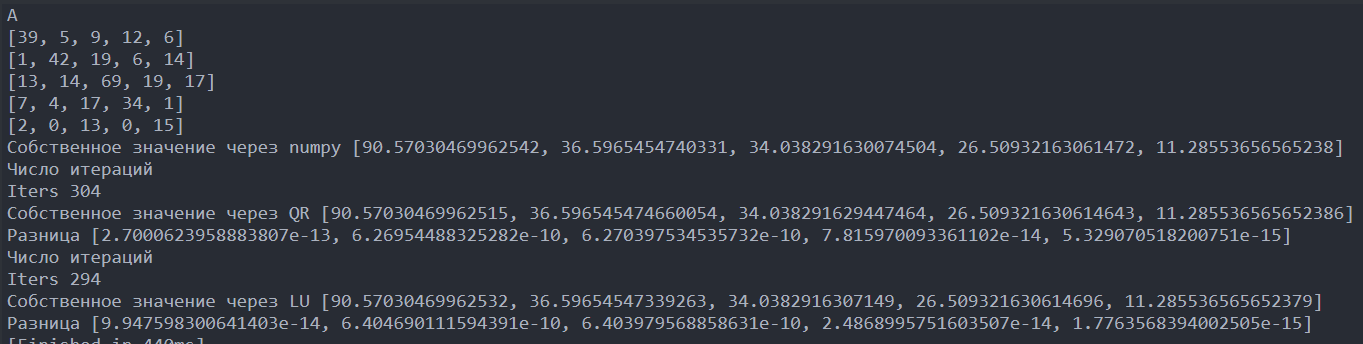
**Первый тест (n=3)**



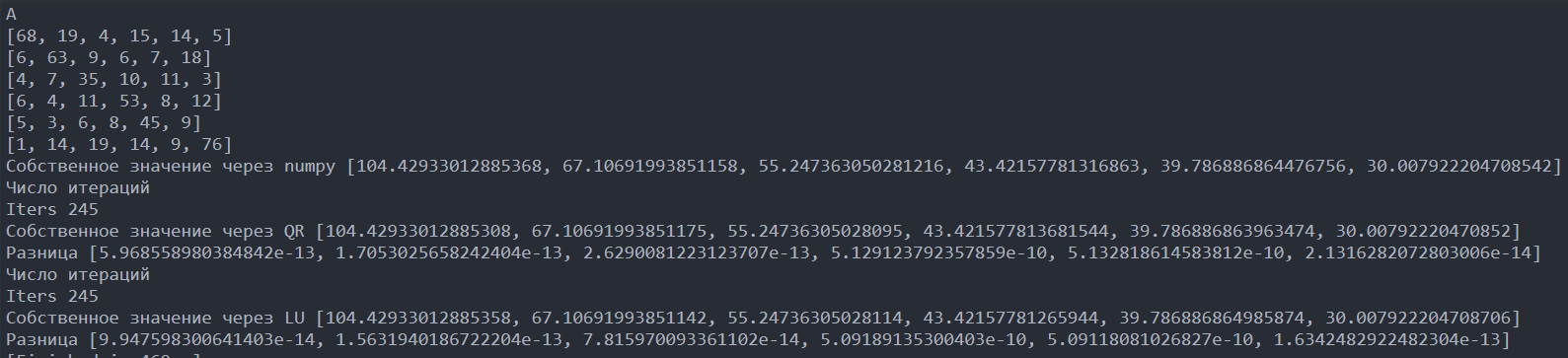
**Второй тест (n=4)**



**Третий тест (n=5)**



**Четвертый тест (n=6)**

****

# **Заключение**

Таким образом нами были реализованы методы нахождения собственных значений при помощи LU и QR разложений на языке Python используя библиотеку Numpy для проверки результатов. Сравнив результаты тестов видно, что при одинаковых вводных данных и точности, LU разложению необходимо меньшее количество итераций.

В результате работы над курсовым проектом мы приобрели практические навыки владения:

* современными численными методами решения задач линейной алгебры;
* основами алгоритмизации для численного решения задач линейной алгебры на одном из языков программирования;
* инструментальными средствами, поддерживающими разработку программного обеспечения для численного решения задач линейной алгебры;

а также навыками представления итогов проделанной работы в виде отчета, оформленного в соответствии с имеющимися требованиями, с привлечением современных средств редактирования и печати.

# **Список использованных источников**

Фаддеев Л. К. Вычислительные методы линейной алгебры / Л. К. Фаддеев, В. Н. Фаддеева. – М.: Физматгиз, 1963.- 656 с

# **Приложения**

def hessel\_rot(matrix):

matrix = [ r[:] for r in matrix]

for j in range(len(matrix)):

for i in range(j+2, len(matrix)):

if(j > len(matrix) - i + 1):

break

c = matrix[j + 1][j]/(matrix[j + 1][j]\*\*2 + matrix[i][j]\*\*2)\*\*0.5

s = matrix[i][j] /(matrix[j + 1][j]\*\*2 + matrix[i][j]\*\*2)\*\*0.5

T = ones(len(matrix))

T[i][i] = c

T[j + 1][j + 1] = c

T[i][j+1] = s

T[j+1][i] = -s

matrix = dot(dot(transpose(T),matrix),T)

return matrix

Листинг 1. Привидение к матрице Хессенберга

**def find\_self(A, func\_of\_razl, eps = 1e-6):**

**A = [r[:] for r in hessel\_rot(A)]**

**counter = 0**

**old\_self = [np.inf]\*len(A)**

**self\_z = [A[i][i] for i in range(len(A))]**

**while(vector\_norm\_1(plus(old\_self, mul(self\_z, -1))) > eps):**

**counter+=1**

**old\_self = self\_z[:]**

**B,C = func\_of\_razl(A)**

**A = dot(C,B)**

**self\_z = [A[i][i] for i in range(len(A))]**

**self\_z.sort()**

**self\_z.reverse()**

**print("Iters", counter)**

**return self\_z**

Листинг 2. Реализация нахождения собственных значений

ЗАДАНИЕ НА КУРСОВОЙ ПРОЕКТ

по дисциплине «Вычислительная математика»

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ФИО

Тема курсового проекта

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Формулировка задания:

1. Изучить предложенный метод.
2. Разработать и отладить алгоритм решения.
3. Провести вычислительные эксперименты.
4. Сделать выводы.

Руководитель проекта\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

ДАТА ВЫДАЧИ задания\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

СРОК ВЫПОЛНЕНИЯ задания\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Задание получил\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_