

Университет ИТМО

Факультет программной инженерии и компьютерной техники

Дисциплина «Вычислительная математика»

Отчет

По лабораторной работе №2

Выполнил:

Терновский И.Е

Преподаватель:

Перл О.В

Санкт-Петербург, 2023 г

Описание метода

Метод Основан на приведении матрицы системы к треугольному виду так, чтобы ниже ее главной диагонали находились только нулевые элементы.

Прямой ход метода Гаусса состоит в последовательном исключении неизвестных из уравнений системы. Сначала с помощью первого уравнения исключается x_1 из всех последующих уравнений системы. Затем с помощью второго уравнения исключается x_2 из третьего и всех последующих уравнений и т. д.

Обратный ход метода Гаусса состоит в последовательном вычислении искомых неизвестных: решая последнее уравнение, находим единственное в этом уравнении неизвестное x_n . Далее, используя это значение, из предыдущего уравнения вычисляем x_{n-1} и т. д. Последним найдем x_1 из первого уравнения.

Класс, реализовывающий сам метод

```
class Solution:
    isSolutionExists = True
    errorMessage = ""

    @staticmethod
    def gaussian_eliminate(matrix):
        """
        Метод для преобразования матрицы к треугольному виду

        :param matrix: исходная матрица
        :return: преобразованная треугольная матрица
        """
        n = len(matrix)
        for i in range(n):
            pivot = matrix[i][i]
            if pivot == 0:
                # Если на главной диагонали находится 0, то метод не применим,
                # выставляем флаг и выходим из метода
                Solution.isSolutionExists = False
                Solution.errorMessage = "The system has no roots of equations or has
an infinite set of them."
                return matrix
            for j in range(i + 1, n):
                ratio = matrix[j][i] / pivot
                matrix[j] = [matrix[j][k] - ratio * matrix[i][k] for k in range(n +
1)]
        return matrix

    @staticmethod
    def back_substitute(matrix):
        """
        Метод для обратной подстановки и нахождения корней системы

        :param matrix: треугольная матрица
```

```

        :return: корни системы уравнений
        """
        n = len(matrix)
        x = [0]*n
        for i in range(n - 1, -1, -1):
            x[i] = (matrix[i][n] - sum(matrix[i][j] * x[j] for j in range(i + 1, n)))
/ matrix[i][i]
        return x

    @staticmethod
    def calculate_residuals(matrix, x):
        """
        Метод для вычисления невязок системы

        :param matrix: исходная матрица
        :param x: корни системы
        :return: невязки системы
        """
        n = len(matrix)
        residuals = [0]*n
        for i in range(n):
            residuals[i] = matrix[i][n] - sum(matrix[i][j] * x[j] for j in range(n))
        return residuals

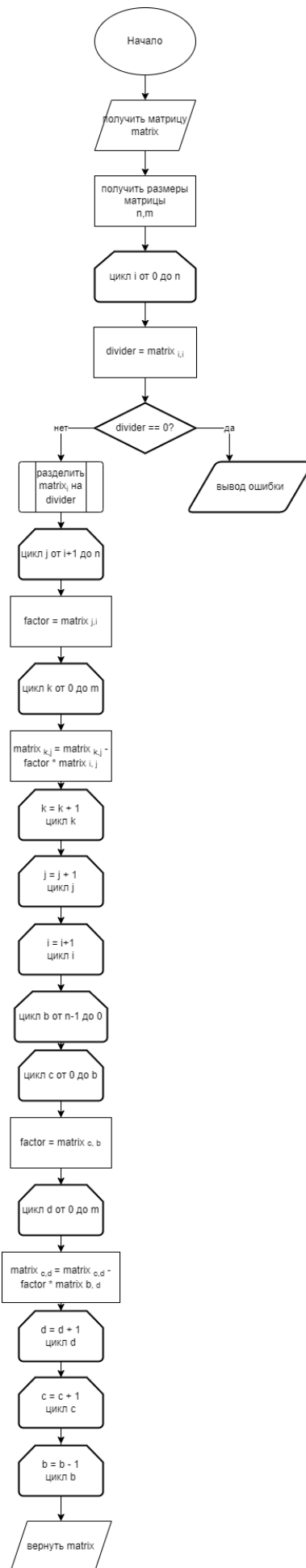
    @staticmethod
    def solveByGauss(n, matrix):
        """
        Основной метод для решения системы уравнений методом Гаусса

        :param n: размерность матрицы (unused)
        :param matrix: исходная матрица
        :return: корни системы уравнений и невязки или сообщение об ошибке если метод не
применим
        """
        matrix_copy = matrix.copy()
        matrix = Solution.gaussian_eliminate(matrix)
        if Solution.isSolutionExists:
            x = Solution.back_substitute(matrix)
            residuals = Solution.calculate_residuals(matrix_copy, x)
            return x + residuals
        else:
            return Solution.errorMessage

if __name__ == '__main__':
    try:
        n = int(input("Enter the number of equations: "))
        matrix = []
        for i in range(n):
            matrix.append(list(map(float, input().split())))
    except ValueError:
        print("Invalid input")
        exit(-1)
    sol = Solution.solveByGauss(n, matrix)
    if Solution.isSolutionExists:
        print("Roots of the system of equations: ", sol[:n])
        print("Residuals of the system of equations: ", sol[n:])
    else:
        print(sol)

```

Блок схема метода:



Пример работы

- 1)

```
Enter the number of equations: 3
1 2 3 1
1 0 3 1
1 2 3 1
The system has no roots of equations or has an infinite set of them.
```
- 2)

```
Enter the number of equations: 3
2 -3 4 6
5 -3 2 3
3 -1 2 2
Roots of the system of equations: [-0.2999999999999998, -0.7999999999999995, 1.0500000000000003]
Residuals of the system of equations: [0.0, 0.0, -4.440892098500626e-16]
```
- 3)

```
Enter the number of equations: 2
3 2 18
1 -1 2
Roots of the system of equations: [4.3999999999999995, 2.4000000000000004]
Residuals of the system of equations: [0.0, 8.881784197001252e-16]
```
- 4)

```
Enter the number of equations: 4
1 2 3 4 30
2 -1 1 3 4
3 2 -1 1 5
4 3 2 -1 6
Roots of the system of equations: [-2.4108108108108084, 5.162162162162163, 2.0648648648648638, 3.9729729729729732]
Residuals of the system of equations: [0.0, -3.552713678800501e-15, -1.0658141036401503e-14, -1.0658141036401503e-14]
```
- 5)

```
Enter the number of equations: 4
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1
The system has no roots of equations or has an infinite set of them.
```

Вывод:

Метод Гаусса позволяет решать СЛАУ с хорошей точностью, в чем можно убедиться из примеров работы. К тому же метод Гаусса достаточно прост в понимании и достаточно эффективен для решения СЛАУ в которых отсутствует нуль на главной диагонали. Метод неприменим в случаях больших матриц и в случаях когда матрица разрежена, в этих случаях следует применять другой метод. Например, в сравнении с методом итераций, метод Гаусса будет более эффективен для небольших и средних систем.

Алгоритмическая сложность n^3 , так как основная часть вычислений происходит в момент приведения матрицы к треугольному виду, для чего требуется 3 вложенных цикла.

Как и в большинстве других методов, численная ошибка накапливается в следствии неточности вычислений компьютера (переполнение), при этом из примеров выше можно заметить, что даже в таком случае точность достаточно высокая.