

Университет ИТМО

Факультет программной инженерии и  
компьютерной техники  
Направление подготовки 09.03.04 Программная  
инженерия  
Дисциплина «Вычислительная математика»

**Отчет  
по лабораторной работе №1  
Вариант 25**

Выполнил:

Туляков Евгений Р32101

Преподаватель:

Малышева Т.А

Санкт-Петербург, 2023 г.

## Цель работы

Разработать программу для подсчета корней СЛАУ.

Для прямых методов должно быть реализовано:

- Вычисление определителя
- Вывод треугольной матрицы (включая преобразованный столбец В)
- Вывод вектора неизвестных:  $x_1, x_2, \dots, x_n$
- Вывод вектора невязок:  $r_1, r_2, \dots, r_n$

## Описание метода

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

## Код программы

<https://github.com/youngpopeugene/ComputationalMath/tree/main/lab1>

## Функция, реализовывающая сам метод

```
public static double[][] calculateTriangleMatrix(double[][] matrix) {
    int n = matrix.length;

    for(int i = 0; i < n; i++) {
        System.out.println(Printer.getBlueText("Iteration № " + (i+1)));

        Integer point = findMaxColumnElement(matrix, i);

        if (point == null) return null;

        if (point != i) System.out.println(Printer.getYellowText("Swapping lines №" + (point + 1) + " and № " + (i+1)));
        else System.out.println(Printer.getYellowText("No swapping required"));

        for (int j = i; j <= n; j++) {
            double temp = matrix[i][j];
            matrix[i][j] = matrix[point][j];
            matrix[point][j] = temp;
        }

        System.out.println(Printer.getYellowText("After swapping:"));
        Printer.printMatrix(matrix);

        for (int k = n; k >= i; k--)
            matrix[i][k] = matrix[i][k] / matrix[i][i];

        for (int k = i + 1; k < n; k++)
            for (int j = n; j >= i; j--) {
                matrix[k][j] -= matrix[k][i] * matrix[i][j];
            }

        System.out.println(Printer.getYellowText("After matrix transformation №" + (i+1)));
        Printer.printMatrix(matrix);
    }

    return matrix;
}
```

## Результат работы программы

GAUSS METHOD with the choice of the main element by columns  
If you want to stop program execution type \exit in console

Enter type of file1: keyboard [k] or file [f]

f

Enter name of file:

file1

Initial matrix:

2,00	3,00	-1,00	7,00
1,00	-1,00	6,00	14,00
6,00	-2,00	1,00	11,00

Iteration № 1

Max column element: 6,00

Swapping lines №3 and № 1

After swapping:

6,00	-2,00	1,00	11,00
1,00	-1,00	6,00	14,00
2,00	3,00	-1,00	7,00

After matrix transformation №1

1,00	-0,33	0,17	1,83
0,00	-0,67	5,83	12,17
0,00	3,67	-1,33	3,33

Iteration № 2

Max column element: 3,67

Swapping lines №3 and № 2

After swapping:

1,00	-0,33	0,17	1,83
0,00	3,67	-1,33	3,33
0,00	-0,67	5,83	12,17

After matrix transformation №2

1,00	-0,33	0,17	1,83
0,00	1,00	-0,36	0,91
0,00	0,00	5,59	12,77

Iteration № 3

Max column element: 5,59

No swapping required

After swapping:

1,00	-0,33	0,17	1,83
0,00	1,00	-0,36	0,91
0,00	0,00	5,59	12,77

After matrix transformation №3

1,00	-0,33	0,17	1,83
------	-------	------	------

0,00	1,00	-0,36	0,91
0,00	0,00	1,00	2,28

Triangle matrix:

1,00	-0,33	0,17	1,83
0,00	1,00	-0,36	0,91
0,00	0,00	1,00	2,28

Matrix determinant:

1,00

Solutions for SLAE:

2,03	1,74	2,28
------	------	------

Residual vector:

-0,00	0,00	0,00
-------	------	------

Process finished with exit code 0

## Вывод

Во время выполнения лабораторной работы я изучил работу прямого метода Гаусса с выбором главного элемента по столбцу.

Основной недостаток прямого метода – хранение всей матрицы в памяти. Также метод не учитывает количество нулевых элементов, в результате чего проводятся лишние арифметические операции. Из-за того, что результаты вычисления используются повторно, накапливается вычислительная погрешность.

При решении СЛАУ методом Гаусса может получиться большая погрешность из-за использования маленьких ведущих элементов. Выбор главного максимального элемента позволяет избежать этого.