

**Федеральное государственное автономное образовательное  
учреждение высшего образования «Национальный  
исследовательский университет ИТМО»**

**Факультет программной инженерии и компьютерной техники**

## **Лабораторная работа №1**

**Вариант 2**

Выполнил:

Брагин Роман Андреевич

Проверил:

Рыбаков Степан Дмитриевич

г. Санкт-Петербург

2025

## Цель работы:

Реализовать метод Гаусса для решения СЛАУ, позволяя вводить данные с клавиатуры или из файла. Программа должна вычислять определитель, приводить матрицу к треугольному виду, находить вектор неизвестных и вектор невязок, а также сравнивать результат с библиотечными методами.

- Вычисление определителя,
- Вывод треугольной матрицы (включая преобразованный столбец В),
- Вывод вектора неизвестных:  $x_1, x_2, \dots, x_n$ ,
- Вывод вектора невязок:  $r_1, r, \dots, r_n$ ,
- Используя библиотеки, найти решение задачи и определитель. Сравнить полученные результаты и объяснить их сходство/различие. Объяснение должно быть отражено в отчёте.

## Описание метода:

Метод основан на приведении матрицы системы к треугольному виду так, чтобы ниже ее главной диагонали находились только нулевые элементы. У этого метода существует 2 хода:

Прямой: заключается в последовательном исключении неизвестных из уравнений. Сначала исключается  $x_1$  из всех последующих уравнений, затем  $x_2$  — из следующих и так далее. Процесс продолжается, пока последнее уравнение не останется с одним неизвестным  $x_n$ , что приводит матрицу к треугольному виду.

Обратный: включает последовательное нахождение неизвестных: сначала вычисляется  $x_n$  из последнего уравнения, затем  $x_{n-1}$  из предыдущего и так далее, пока не будет найдено  $x_1$ . Существует несколько вариантов реализации метода.

## Листинг программы:

[Ссылка на репозиторий с Гитхаба](#)



## Часть с реализацией метода:

```

func Gauss(A [][]float64, b []float64) ([]float64, float64, [][]float64, []float64) {
    n := len(A)
    det := 1.0

    // Прямой ход
    for i := 0; i < n; i++ {
        if math.Abs(A[i][i]) < 1e-9 {
            maxRow := i
            for k := i + 1; k < n; k++ {
                if math.Abs(A[k][i]) > math.Abs(A[maxRow][i]) {
                    maxRow = k
                }
            }

            if math.Abs(A[maxRow][i]) < 1e-9 {
                panic("Система несовместна или имеет бесконечно много решений")
            }

            // Перестановка строк
            A[i], A[maxRow] = A[maxRow], A[i]
            b[i], b[maxRow] = b[maxRow], b[i]
            det = -det // Меняем знак определителя
        }

        det *= A[i][i]
        for k := i + 1; k < n; k++ {
            factor := A[k][i] / A[i][i]
            b[k] -= factor * b[i]
            for j := i; j < n; j++ {
                A[k][j] -= factor * A[i][j]
            }
        }
    }

    // Обратный ход
    x := make([]float64, n)
    for i := n - 1; i >= 0; i-- {
        sum := b[i]
        for j := i + 1; j < n; j++ {
            sum -= A[i][j] * x[j]
        }
        x[i] = sum / A[i][i]
    }

    return x, det, A, b
}

```

## Пример работы (из презентации):

Введите метод ввода данных:

Клавиатура (1), Файл (2) : 1

Размерность матрицы ( $n \leq 20$ ): 3

Введите коэффициенты матрицы A:

10 -7 0  
-3 2 6  
5 -1 5  
Введите столбец В:  
7 4 6

----- Вывод -----

Треугольная матрица :  
10.00	-7.00	0.00	7.00
0.00	-0.10	6.00	6.10
0.00	0.00	155.00	155.00
Вектор невязок: [0 0 0]

----- Вывод моей программы -----

Определитель: -155.00000000000003  
Вектор неизвестных: [-9.325873406851315e-15 -1.0000000000000133 0.9999999999999998]

----- Вывод библиотеки -----

Определитель : -155.00000000000003  
Вектор неизвестных: [-9.325873406851315e-15 -1.0000000000000133 0.9999999999999998]

Решение из лекции: (0, -1, 1) что  $\sim$  (-0.00000000000000933, -1, 1) что является верным

## Вывод:

В ходе работы я реализовал метод Гаусса для решения СЛАУ, что позволило мне глубже понять его принципы и применение на практике.