

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ  
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

**Факультет прикладной математики и информатики**

Лабораторная работа №1

**РЕШЕНИЕ СИСТЕМ ЛИНЕЙНЫХ УРАВНЕНИЙ МЕТОДОМ  
ГАУССА**

**С ВЫБОРОМ ГЛАВНОГО ЭЛЕМЕНТА ПО СТОЛБЦУ**

**Преподаватель:** Полевиков Виктор Кузьмич  
доцент кафедры вычислительной математики

**Студент:** Болтач Антон Юрьевич  
2 курс 9 группа

**Минск 2017**

## Постановка задачи

Рассмотрим СЛАУ вида:

$$A \cdot x = f$$

Задана матрица  $A$  размером 20 на 20. Элементами которой являются вещественные числа на отрезке от -100 до 100 с точностью до двух знаков после запятой:

**Матрица  $A$ :**

-88.45	80.16	-71.89	-39.5	64.93	-23.9	20.69	88.66	55.77	36.5	52.37	7.72	-63.55	20.82	96.99	49.33	57.13	73.15	-34.23	-2.83
90.51	85.32	50.22	-5.31	57.71	77.81	87.26	75.84	51.73	32.31	-24.68	-24.73	5.5	-17.9	-8.09	31.51	39.15	10.22	-10.0	93.25
93.86	-75.35	87.0	-37.83	67.88	-18.61	-99.74	97.83	-61.14	-1.02	42.6	76.52	13.23	-36.66	44.76	-3.08	-83.56	63.77	-65.07	62.33
-50.37	-8.1	-2.71	9.24	2.71	-83.78	99.94	-58.73	49.38	-16.52	-92.27	-37.61	-23.21	47.72	85.0	-77.72	-11.36	48.59	-97.23	-95.32
-57.64	-25.82	35.68	-87.95	-75.36	-66.42	34.5	-3.44	41.91	-97.16	61.39	-31.65	-0.69	75.34	43.12	61.19	-36.37	-33.71	63.9	33.86
27.57	-2.03	46.63	1.78	5.71	-70.26	-22.38	74.47	33.71	58.65	-37.88	-81.04	-9.48	-82.35	-77.74	22.47	-63.58	48.23	68.88	12.54
-30.82	4.41	-44.27	-65.0	-47.44	-67.61	91.28	-36.54	-22.6	-11.36	-20.69	61.27	70.24	48.87	47.39	97.42	61.26	-61.31	4.66	40.52
-25.14	-40.96	-30.74	-56.63	0.08	-78.71	53.58	-10.89	-61.2	-91.43	90.64	-21.15	-2.1	70.37	-33.17	12.92	-36.56	-48.36	-64.9	32.82
-19.48	46.23	12.23	-21.82	-77.79	53.47	54.25	75.17	-24.01	21.9	-42.83	49.99	8.84	-83.25	43.84	23.42	-82.79	19.76	62.69	-87.92
-72.04	-36.43	-89.86	-15.91	-73.61	-72.64	18.98	43.98	94.28	-14.5	30.15	62.06	3.78	-95.81	7.12	-22.24	-99.11	-72.35	-17.57	-58.98
75.87	89.74	11.16	-22.03	-25.6	83.57	-12.47	13.5	57.33	-34.97	-99.68	2.24	29.28	27.81	0.09	53.89	98.13	66.42	76.31	-75.62
-52.64	8.01	21.45	7.32	33.96	3.81	63.86	-53.08	76.99	23.73	65.89	73.57	-64.42	-7.8	62.49	-57.78	76.8	8.94	64.37	3.06
44.2	-61.82	4.74	-82.94	-35.47	43.99	-52.11	11.39	-22.72	-69.86	90.8	79.02	-95.28	-14.81	-52.36	80.54	-97.16	77.22	39.12	75.63
72.71	-43.25	-71.41	-1.47	-79.53	83.9	-54.02	25.68	-89.76	80.3	54.99	-41.33	-28.39	-18.46	27.57	-2.75	65.46	90.96	-17.0	-41.9
-51.52	-39.97	-66.14	-38.42	-50.3	-59.86	14.43	-50.08	57.9	-75.53	-20.91	-18.0	-97.08	-80.22	83.99	68.59	-7.04	-27.81	-32.3	-95.53
-97.01	76.64	37.98	-95.83	-53.3	72.36	30.98	-45.36	69.26	24.47	50.16	-31.39	-46.97	15.68	30.05	-79.6	75.27	89.66	-21.49	98.57
57.66	-42.36	6.98	-69.53	-53.41	-33.92	97.39	-83.73	18.82	-26.87	-94.15	48.33	-76.52	-89.19	10.03	-96.46	48.99	-83.22	50.48	-97.63
-60.65	66.13	-85.02	-97.87	-34.86	-17.19	-34.39	-84.45	10.62	63.14	28.71	-72.88	30.7	77.46	62.62	15.46	40.67	87.21	10.48	34.9
46.21	-87.17	66.45	-9.81	-80.51	-76.94	-48.11	-67.54	95.58	94.14	30.44	6.05	-74.47	-86.16	-79.99	-31.3	59.9	-94.21	-10.53	-41.59

-14.96 53.4 -5.81 82.84 59.2 -12.3 -35.43 -8.01 0.09 12.0 -4.91 69.77 -24.1 -85.88 -  
58.77 -24.09 30.18 -97.89 -59.37 -16.12

**Транспонированный столбец свободных коэффициентов  $f$ :**

5.73 16.09 -24.98 -61.57 33.35 38.22 -14.18 3.65 87.18 20.16 97.64 17.35 -74.83 38.62 -  
21.32 -21.88 -61.54 22.27 75.36 -78.71

**Для Матрицы  $A$  и столбца свободных коэффициентов  $f$ , найти:**

- 1) Найти решение системы линейных алгебраических уравнений методом Гаусса.
- 2) Вычислить определитель матрицы  $A$ .
- 3) Найти матрицу, обратную матрице  $A$ , используя метод Гаусса.
- 4) Определить меру обусловленности матрицы  $A$ .

## Краткая теория и алгоритм решения

Система линейных уравнений может иметь одно решение, бесконечно много решений или же быть несовместной (не иметь решений). Не все методы решения СЛАУ могут справиться с вторым случаем, когда система имеет бесконечно много решений. Например, метод Крамера и матричный метод не применимы, но метод Гаусса вполне можно использовать.

Алгоритм можно условно разделить на два этапа:

- Прямой ход
- Обратный ход

В первом этапе образуются нули ниже или выше главной диагонали, за счет использования **элементарных преобразований матрицы**. При этом матрица  $A$  преобразуется к треугольной. На втором этапе находят неизвестные начиная с конца.

### Последовательное исключение(Прямой ход)

Прямой ход метода Гаусса имеет своей целью приведение расширенной матрицы системы к ступенчатому виду. Есть ли решения у системы (система совместна) или же решений нет (система несовместна) выяснится именно здесь, в конце прямого хода метода Гаусса.

Предположим, что  $a_{11} \neq 0$ , если нет, то переставим местами первую и  $k$ -тую строки, где  $a_{k1} \neq 0$  и выбираем его в качестве ведущего элемента. Разделив первое уравнение на  $a_{11}$ , получим:

$$x_1 + c_{12}x_2 + \dots + c_{1n}x_n = q_1(1), \text{ где } c_{1j} = \frac{a_{1j}}{a_{11}}, j = \overline{2, n}, q_1 = \frac{b_1}{a_{11}}$$

С помощью уравнения (1) исключим  $x_1$  из оставшихся уравнений системы:

С помощью формул(1) исключим  $x_1$  из всех оставшихся уравнений системы. В результате получим следующую систему:

$$\begin{cases} a_{22}^{(1)}x_2 + a_{23}^{(1)}x_3 + \dots + a_{2n}^{(1)}x_n = b_2^{(1)} \\ \text{-----} \\ a_{n2}^{(1)}x_2 + a_{n3}^{(1)}x_3 + \dots + a_{nn}^{(1)}x_n = b_n^{(1)} \end{cases} \quad (2), \text{ где } \begin{cases} a_{ij}^{(1)} = a_{ij} - c_{1j}a_{i1} \\ b_i^{(1)} = b_i - q_1a_{i1} \end{cases}, i, j = \overline{2, n}$$

Аналогично исключая последовательно неизвестные  $x_2, x_3, \dots, x_{n-1}$ , окончательно приходим к системе уравнений с матрицей вида:

$$\begin{cases} x_1 + c_{12}x_2 + \dots + c_{1n}x_n = q_1 \\ \quad \quad \quad x_2 + \dots + c_{2n}x_n = q_2 \\ \text{-----} \\ \quad \quad \quad \quad \quad x_n = q_n \end{cases} \quad (3)$$

Получение системы (3) составляет прямой ход метода Гаусса.

В системе (3) имеем:  $a_{kj}^0 = a_{kj}$ ,  $k, j = \overline{1, n}$ ,  $b_k^0 = b_k$ ,  $k = \overline{1, n}$

$$c_{kj} = \frac{a_{kj}^{(k-1)}}{a_{kk}^{(k-1)}}, j = \overline{k+1, n}, k = \overline{1, n-1}$$

$$a_{ij}^k = a_{ij}^{(k-1)} - a_{ik}^{(k-1)} c_{kj}, i, j = \overline{k+1, n}, k = \overline{1, n-1}$$

$$q_k = \frac{b_k^{(k-1)}}{a_{kk}^{(k-1)}}, k = \overline{1, n}, b_i^k = b_i^{(k-1)} - a_{ik}^{(k-1)} q_k, i = \overline{k+1, n}, k = \overline{1, n-1}$$

$a_{kk}^{(k-1)}$  – ведущий элемент на k-м этапе исключения.

### Обратный ход

Обратный ход состоит в последовательном нахождении неизвестных  $x_n, x_{n-1}, \dots, x_1$  из системы (3). Тогда имеем следующие формулы:

$$\begin{cases} x_n = f_n \\ x_k = (b_k^{(k-1)} - \sum_{j=k+1}^n a_{kj}^{(k-1)} x_j) / a_{kk}^{(k-1)}, k = \overline{n-1, 1} \end{cases}$$

**Рассмотрим метод Гаусса с выбором главного элемента по столбцу.** Отличие данного метода Гаусса от схемы единственного деления заключается в том, что на k-м шаге исключения в качестве главного элемента выбираем максимальный по модулю коэффициент при неизвестной  $x_k$  в уравнениях с номерами  $i = \overline{k', n}$ . Затем строка расширенной матрицы, соответствующая главному элементу, меняется местами с k-й строкой данной матрицы и производится перенумерация коэффициентов при неизвестных.

После этой перестановки исключение неизвестных производят так же, как и в схеме единственного деления.

### Вычисление определителя:

Так как мы сводим матрицу A к треугольному виду A' при помощи элементарных преобразований двух видов (1. Перестановка двух строк; 2. Из i-й строки вычесть j-ю, умноженную на  $\alpha$ ), то  $\det A = (-1)^m \cdot \det A' = (-1)^m \cdot a_{11}^{(0)} \cdot a_{22}^{(1)} \cdot \dots \cdot a_{nn}^{(n-1)}$ , где m – количество перестановок, осуществляемых в процессе исключения,  $a_{kk}^{(k-1)}$  – ведущие элементы метода Гаусса ( $k = \overline{1, n'-1}$ ).

### Вычисление обратной матрицы:

Пусть A' – обратная матрица, E – единичная матрица, тогда верно, что:

$$A \cdot A' = E$$

Отсюда обратную матрицу можно вычислить, решив  $n$  систем вида:  $Ax^{(j)} = e^{(j)}$ , где  $x^{(j)} = (x_{1j}, x_{2j}, \dots, x_{nj})^T$ ,  $e^{(j)} = (e_{1j}, e_{2j}, \dots, e_{nj})^T$ ,  $j = 1, n$ ,  $e_{jj} = 1$ ,  $e_{ij} = 0$  ( $i \neq j$ )

### Число обусловленности:

Число обусловленности исходной матрицы ( $\nu(A)$ ) находим в результате произведения нормы матрицы  $A$  на норму обратной матрицы  $A^{-1}$ :

$$\nu(A) = \|A\| \cdot \|A^{-1}\|$$

Для нахождения нормы воспользуемся формулой кубической нормы матрицы (т.е. норма матрицы, подчиненная кубической норме векторов). Она равна максимальной сумме модулей элементов в каждой строке:

$$\|A\| = \max_{1 \leq i \leq n} \sum_{j=1}^n |a_{ij}|$$

### Листинг программы на языке программирования Java

```
import java.util.Random;

public class Gauss {

    public static void swap(double x, double y) {

        double tmp = x;

        x = y;

        y = tmp;

    }

    public static void output(double [][]mat, int n, int m){

        for (int i = 0; i < n; i++){

            System.out.print("");

            for(int j = 0; j < m; j++)

                System.out.print(mat[i][j] + " ");

            System.out.println("");

        }

    }

    public static void output(double []mat, int n){

        for (int i = 0; i < n; i++)

            System.out.print(mat[i] + " ");

        System.out.println("");

    }

}
```

```

//Метод Гаусса

public static int GaussMethod(double[][] matrix, double[] Y, double[] X,
int n){

    double workMatrix[][] = new double[n][n];

    //Создаем матрицу
    for (int i = 0; i < n; i++)
        for(int j = 0; j < n; j++)
            workMatrix[i][j] = matrix[i][j];

    //Проверяем полученную матрицу на возможность решений
    //А так же подсчитываем количество перестановок
    int count = 0;
    double max;
    int index;
    double eps = 0.00001;
    int amount = 0;
    //Выбираем Максимальный Элемент и ставим на нужное место
    while(amount < n) {
        max = Math.abs(matrix[amount][amount]);
        index = amount;
        for(int i = amount + 1; i < n; i++) {
            if(Math.abs(matrix[i][amount]) > max) {
                max = Math.abs(matrix[i][amount]);
                index = i;
            }
        }
        //Нет нулевых диагональных Элементов
        if(max < eps) {
            System.out.print("Решение нет, нулевой столбец");
            System.out.println(index + "Матрицы A");
            return 1;
        }
        for(int j = 0; j < n; j++)
            swap(matrix[amount][j], matrix[index][j]);
        swap(Y[amount], Y[index]);
    }
}

```

```

        //Вспоминаем про перестановки
        if(index != amount)
            count++;
        amount++;
    }
    //Считаем определитель приводя к верхнетреуг. Матрице
    double det = 1;
    for(int i = 0; i < n; i++) {
        for(int j = 0; j < n; j++) {
            if(j > i) {
                double element = matrix[j][i] / matrix[i][i];
                for(int k = 0; k < n; k++)
                    matrix[j][k] -= matrix[i][k] * element;
                Y[j] -= Y[i] * element;
            }
        }
        det *= matrix[i][i];
    }
    System.out.println("Определитель det(A): " + det);
    System.out.println("");
    for(int i = 0; i < n; i++) {
        double element = matrix[i][i];
        for(int j = 0; j < n; j++)
            matrix[i][j] = matrix[i][j] / element;
        Y[i] = Y[i] / element;
    }
    for(int k = n - 1; k >= 0; k--){
        X[k] = Y[k];
        for(int i = 0; i < k; i++)
            Y[i] = Y[i] - matrix[i][k] * X[k];
    }
    for(int i = 0; i < n; i++) {
        for(int j = 0; j < n; j++)
            matrix[i][j] = workMatrix[i][j];
    }
    return 0;

```



```

    }

    public static void main(String[] args) {

        int n = 20; //Размер Матрицы
        double matrix[][] = new double[n][n];
        double reverse[][] = new double[n][n];
        double Y[] = new double[n];

        double Unit[][] = new double[n][n];
        for(int i = 0; i < n; i++){
            for (int j = 0; j < n; j++)
                Unit[i][j] = 0.00;
            Unit[i][i] = 1.00;
        }

        Random r = new Random();
        for (int i = 0; i < n; i++) {
            for (int j = 0; j < n; j++) {
                matrix[i][j] = -100 + (200) * r.nextDouble();
                matrix[i][j] = Math rint(matrix[i][j] * 100.0) / 100.0;
            }
            Y[i] = -100 + (200) * r.nextDouble();
            Y[i] = Math rint(Y[i] * 100.0) / 100.0;
        }

        System.out.println("Матрица 20x20 полученная рандомом на интервале (-100;100): ");
        output(matrix, n, n);
        System.out.println("Столбец свободных коэффициентов f: ");
        output(Y, n);

        double x[] = new double[n];
        //Проверяем на работоспособность метода
        if (GaussMethod(matrix, Y, x, n) == 1)
            return;

        System.out.println("Решения для Матрицы 20x20, то есть корни уравнений");
        output(x, n);
    }
}

```

```

for(int i = 0; i < n; i++) {
    GaussMethod(matrix, Unit[i], x, n);
    for(int j = 0; j < n; j++)
        reverse[j][i] = x[j];
}

System.out.println("Обратная Матрица 20x20:");
output(reverse, n, n);

for(int i = 0; i < n; i++){
    for(int j = 0; j < n; j++) {
        Unit[i][j] = 0.00;
        for(int l = 0; l < n; l++)
            Unit[i][j] += matrix[i][l] * reverse[l][j];
    }
}

System.out.println("Матрица, полученная по формуле: A*A^T" );
output(Unit, n, n);

//Считаем меру обусловленности
double maxFirst = Integer.MIN_VALUE, maxSecond = Integer.MIN_VALUE;
double summaryFirst = 0, summarySecond = 0;
for(int i = 0; i < n; i++) {
    for(int j = 0; j < n; j++) {
        summaryFirst += Math.abs(matrix[i][j]);
        summarySecond += Math.abs(reverse[i][j]);
    }

    if (summaryFirst > maxFirst)
        maxFirst = summaryFirst;
    if (summarySecond > maxSecond)
        maxSecond = summarySecond;
}

```

```

        System.out.println("Мера обусловленности равна (v(A)) : ");

        System.out.println(maxFirst * maxSecond);

    }

}

```

## Результаты

### Решения:

```

-0.3010846086211623 -0.5019277470482444 0.6472394889443107 -0.8587298919096487
0.7245072237557149 0.695037760416294 0.3900723088744608 -0.09277058186063386
0.43940329411238865 0.35930796074304205 1.0803961004705789 -0.47697990998013884
1.3793323686178263 -0.38408027371430253 -0.4688780132197776 0.601307649718963
0.19476711087824305 0.10624590488564917 0.014668704946639646 -1.490937790213794

```

**Определитель:** -5.849227083227985E44

### Обратная матрица:

```

-0.0027292493687617186 0.005232782343140823 0.001435922365780574 0.0035444238085167532 0.0053964731943286865
1.435311338835979E-4 -0.001298093169085531 -7.922555856554629E-4 -0.00151041960350669 0.001431165338974639
0.0011393359686502269 0.0013515566304155586 0.001901755732106724 0.004455839269979846 -0.0020365182315289668
-0.004135589497085855 0.0015005506990781571 0.004011462084371696 -8.26052186480926E-4 0.005676597533684742

-0.0013950823137605108 0.003206696900324846 2.13780296557624E-4 0.004060553086675161 0.007819989180343663
0.0011746034407324646 -0.001580250930446524 -5.930670431649648E-4 0.001908400660880591 -2.4259000894872413E-4
0.0011418250854570242 2.969421683861635E-4 0.0016699634157488058 0.0033417347401360122 -0.002946766105806111
-0.001665068967683597 7.171623780732058E-4 0.005468949295565135 -0.0019177124529580463 0.011824795522565271

-0.0011752093176307793 -7.642957493846254E-4 0.0019493976090498837 1.1958552474646694E-4 1.3831281162708653E-
4 4.7992460063712994E-5 -2.2314437500409005E-4 0.002029192050642295 0.0037413274080592973 -
0.003622668530224446 8.760103286743169E-4 0.0012261096193292177 -0.0016898012401748968 -0.002069004549220134
0.00116659720359134 0.0016540634097943659 -0.0022682202554057222 -0.0014487206851953273 0.0028117842933886653
-5.167072910144607E-4

-0.003324882250407118 0.001466189611032914 -0.0016121308220760188 0.0054017724578887556 0.006950671220723849
0.0023017340674944308 0.0012033418230521288 -0.005071083392552734 -0.002604372649173262 7.992425382001077E-4 -
0.0010944690023300865 0.002589033321190752 0.0018915903520813166 0.006166925758243284 -0.001631170521675919 -
0.0018635189395374762 -0.002443326812235737 -0.0013346930776569669 -0.003167777970624632 0.008084882578104685

0.002061444888776497 -8.17496846996912E-4 0.001016413605525008 -0.0052050093993994215 -0.009282188797815413 -
0.0011324569593363985 -0.0021898327141840913 0.005060676845980399 9.458390776602555E-4 -0.001818731464469027
3.86716106503857E-4 0.0021159063103991993 -0.002627991444062348 -0.006937926868093676 0.0035662952133037025 -
9.068216760346732E-4 -1.8986205121439813E-4 4.0402200089939165E-4 0.0012227707611852994 -0.007984709716566285

8.188445899809751E-4 2.1655247291779006E-4 -0.0014870243112737832 -0.004750734876187572 -0.00663392838530651 -
0.005196516720101916 -0.0022851497935457007 8.882212232352721E-4 0.0022210656488488084 -6.992059854765005E-4 -
6.026387387562302E-4 -0.001573944015022608 -0.0010842314020529955 -0.00428119334332316 0.002652606340766797
0.0016118829865327203 -0.001000370866226695 -0.0019324594878963662 0.0019215886820522993 -
0.008394399648262156

-0.0013726510801579735 0.00222004661243931 -0.001323946105494961 9.166657633963383E-4 -0.0034583926069924826
0.001113010292403325 0.0016411147011075114 0.004148499689052569 0.002080691077352417 -2.2879839019469958E-5 -
1.5447286347527138E-4 0.0032148352998190833 -1.7310631713666168E-4 -1.4894422390820364E-4 8.437756244974439E-
4 8.677886039992506E-5 -0.001076384241034808 -0.0013193192271889456 1.015194999149392E-4 -
0.0027068961811491056

0.00496709024451763 -9.52315195923005E-4 -1.1989091953069207E-4 -5.356042267784055E-4 0.0030283094639827596
4.827976743348861E-4 1.0172725204521448E-4 -0.0020995931502051975 -0.0014688908319774045
0.0014434201307991974 -2.646029374192184E-4 -0.003738856132366869 -7.46912239413113E-4 0.0017277597701243776 -
0.0031990116506504748 8.768183218384675E-4 0.0019269546142270453 -0.0043631764251448844 -7.703005722105148E-4
-5.032209476079667E-4

-1.6783485703273163E-4 0.0019612834711350375 -2.2630850549096304E-4 1.798920679705546E-4 -
0.0011326236554139966 -0.0013086981147096416 -0.0012411655916630272 -8.404453761469317E-4 -

```

0.0018293863169285204 0.003382466052706052 0.001880049384394452 8.830099261271061E-4 1.4593359250329426E-4 -  
0.002154983023684328 6.484649578452385E-4 -2.1842947494000488E-4 -0.001663594173286574 1.759701088812904E-4  
0.0018400563509388262 -0.003025356212021414

0.0029489116672100045 6.040405417244506E-4 -0.0012879272356564613 -3.084156728866835E-4 -0.002752525904415875  
-0.001930929603991373 1.3196006133680237E-4 -0.0018031875936295495 0.00276422987889775 -0.0014329557382674642  
-0.0033058026644327875 -0.0015027852358068718 4.2611657980086426E-4 -0.0019154373180352911 -  
0.0020989643924393016 -0.00180877368816902 -3.4201371327006666E-6 0.0018667412046074242 0.004034817723213404 -  
0.003405189519834424

-0.003018280909655512 0.0016909503566056193 0.00147390114360844 -0.0010842694523109334 -3.301808686474791E-4  
3.264291669392899E-4 -0.002046087519526594 0.00689895111665322 0.0025538507942699344 8.639386200148991E-4  
0.002058353883716141 0.0062725264810003225 -0.001645078483351462 0.0016175977898587796 0.0015170448494467976  
-0.0010155868032274337 -0.0033980338833296065 0.0024706426561587344 0.0011528480597319103  
0.0011618511773278763

8.47623227465634E-4 -0.0013877490535074656 6.848180172035077E-4 0.002011245312793576 5.025125703208991E-5 -  
0.0011719804733530204 0.0026505843103010068 -0.0018677240865574325 2.1206037038130898E-4  
0.0011322743595464793 9.374183617604501E-4 7.061644025331637E-4 0.003117811555703228 -2.3137510966441798E-5 -  
0.0031431992626711675 -3.6676010228027403E-4 2.984103058981779E-4 -6.235043948411554E-4 1.803026728887079E-4  
0.0025196410848076184

-0.004746856010404005 -8.185539756023253E-4 0.002898201955919579 -0.0033642911112830654 -0.006597160064018054  
9.885661399613261E-4 0.0016764114125336544 0.005994815719624265 0.00142057278531098 0.0025838539790054827  
0.004057465846259708 0.0039532741389195075 -0.004917704971023607 -0.0023665072410120913 0.002932481426867962  
0.001606667315340154 -0.003891803805333928 6.173761992428326E-4 1.4406051064366635E-4 -0.0048959033036308445

0.004324398024901655 -2.983891732688624E-4 -0.002493735807787234 0.002960231296981616 0.004393846671130033 -  
0.004044554692498087 -0.0012992238851322552 -0.0036749508869574673 -0.0010429241464843884 -  
9.310369632741294E-4 -0.0013600803613810723 -0.003953624857644385 0.0027971070662195703 -3.487978911970134E-4  
-0.00589238860626989 -0.002770195558719531 0.001429519285233963 3.051488324093323E-5 0.0016397774486603695 -  
3.848809128027802E-4

-7.467369948210745E-4 0.002350628060482815 0.002152332327035374 0.0015507315583804075 0.006400950341027688 -  
0.0015875178789743203 -5.17908057562409E-4 -0.005055692219020654 -4.983336298232822E-4 -1.3816930125808147E-4  
-0.002460588337172849 7.735828224529226E-4 -0.001074342066238066 0.002907135367452525 9.148501051932373E-4 -  
0.0018149940225149481 0.0010734302303831021 0.0021450120263846397 -0.002203533149031942  
0.003164309178406137

7.849580668657918E-5 5.491104396720761E-4 -6.970815676219763E-4 -0.0013557975412364074 -0.003456727614487536 -  
3.3736369686340544E-4 0.002139213777251929 0.0015496437165795096 0.0024074671871756647 -0.00242021915271586  
5.414442321027025E-4 3.9364459355288286E-4 6.295271609577541E-4 -0.002325657220256278 0.0039217186123403636 -  
0.0010352242554567122 -0.0036778410037218473 -3.354898952909897E-4 0.0024986111824354634 -  
0.002717895913476635

9.053454707955797E-4 -0.0022915387787101208 6.004802886875752E-4 -0.0018931741895230423 -0.001272559104688608  
0.0018272037334801268 0.002834661026569681 0.0013606940442694952 -0.0025520149701723973 -5.638354215285377E-4  
0.0026387827749276126 0.0011184762115124675 -0.0024696569822482495 0.0018034577238939331  
0.001082919333504241 0.002844433026619691 1.2798359418593475E-4 -0.003499909987796119 -2.859461274556264E-4 -  
3.52611216912969E-4

-0.0026020091423236353 -0.001453248697422639 0.0012203949597510136 0.0027035981235110198 -  
0.0029497918484453272 0.004374023660368596 0.0019849244963275328 0.0024332036117583507 -3.8245645944020727E-  
4 6.25554560963385E-4 0.0032503613544608567 0.003984016438290447 0.0013205116533023507 0.001866405320258648  
9.35537086286556E-4 0.0018390048951021918 -0.0027711876024256715 -4.6296204004554493E-4 -  
0.0016409809220438926 2.979926194796817E-4

1.6062719150184023E-4 -0.0011190275378442683 -0.001083480622686913 -0.0018064526353818717  
0.0027464615899179174 0.0020403840751441055 -5.259695126223801E-4 -0.0022851881705598072 -7.432297625019974E-  
4 -1.994954325314318E-4 -4.785656604282884E-4 0.002157918753676525 1.5869883873071615E-4 4.748607759580552E-4  
-0.0014525392420311673 -0.0017146657591787112 0.0021980076056669394 6.30971927097922E-4 -  
0.0020971652135167314 -9.122990470362246E-5

-9.068357527416531E-4 0.0020211746937609126 -3.6870085879938725E-4 0.0017684891405433214 0.005597740842111297  
0.0017784633309606353 0.002762203395909109 -0.006800148363794442 -0.004952177942586644 0.001156339860255449 -  
0.003577792657000389 -0.0019146581165986705 0.002537540518940929 0.0031642272500601158 -0.0019839720413012264  
0.0013305539605427017 0.001791477739361639 -4.468460277763983E-4 -0.0036884456871832255 0.004645222625765426

1.00000000000000000000 -2.42861286636753E-17 -1.8865117801247777E-17 7.719519468096792E-17  
-2.42861286636753E-17 6.678685382510707E-17 -1.734723475976807E-17  
1.5265566588595902E-16 5.898059818321144E-17 -5.681219383824043E-17  
1.214306433183765E-16 2.2551405187698492E-17 6.591949208711867E-17 -  
1.457167719820518E-16 6.678685382510707E-17 -3.7730235602495554E-17 -  
2.168404344971009E-17 -1.1687699419393738E-16 -2.949029909160572E-17  
1.0061396160665481E-16  
  
-5.551115123125783E-16 0.9999999999999997 1.5265566588595902E-16 1.3877787807814457E-  
16 0.0 0.0 1.1102230246251565E-16 -1.1102230246251565E-16 0.0 1.6653345369377348E-16 -  
5.551115123125783E-17 -1.1102230246251565E-16 5.551115123125783E-17  
1.1102230246251565E-16 0.0 9.71445146547012E-17 -5.551115123125783E-17  
3.400058012914542E-16 -1.1102230246251565E-16 5.551115123125783E-17  
  
-4.2327252813834093E-16 -8.326672684688674E-17 0.9999999999999999 0.0 -  
5.551115123125783E-17 -1.249000902703301E-16 -8.326672684688674E-17 -  
2.7755575615628914E-16 0.0 9.71445146547012E-17 -1.6653345369377348E-16 -  
2.498001805406602E-16 1.942890293094024E-16 1.3877787807814457E-16 -9.71445146547012E-  
17 -4.163336342344337E-17 1.6653345369377348E-16 1.4224732503009818E-16  
5.551115123125783E-17 3.885780586188048E-16  
  
3.469446951953614E-16 5.551115123125783E-16 -4.718447854656915E-16 1.0  
9.992007221626409E-16 -2.7755575615628914E-17 -3.885780586188048E-16  
3.3306690738754696E-16 4.440892098500626E-16 9.71445146547012E-17 3.3306690738754696E-  
16 4.440892098500626E-16 9.159339953157541E-16 -5.551115123125783E-16  
6.938893903907228E-16 -4.440892098500626E-16 7.494005416219807E-16  
2.0816681711721685E-16 3.3306690738754696E-16 9.43689570931383E-16  
  
-9.749145934989656E-16 1.0130785099704553E-15 -1.2125717097077882E-15  
6.106226635438361E-16 1.00000000000000018 -8.118505867571457E-16 -9.020562075079397E-16  
-6.106226635438361E-16 6.38378239159465E-16 -9.020562075079397E-17 -  
2.914335439641036E-16 4.85722573273506E-16 2.123301534595612E-15 -1.27675647831893E-15  
1.9290125052862095E-15 -8.534839501805891E-16 1.2004286453759505E-15  
6.071532165918825E-16 1.249000902703301E-16 1.8596235662471372E-15  
  
-6.279698983036042E-16 2.8449465006019636E-16 -2.5673907444456745E-16  
3.5041414214731503E-16 1.1934897514720433E-15 0.9999999999999999 -2.42861286636753E-16  
1.3877787807814457E-16 2.914335439641036E-16 1.2836953722228372E-16  
5.551115123125783E-17 4.3021142204224816E-16 7.28583859910259E-16 2.0816681711721685E-  
17 5.238864897449957E-16 -3.2612801348363973E-16 4.891920202254596E-16  
3.7296554733501353E-16 6.938893903907228E-17 8.534839501805891E-16  
  
-6.245004513516506E-16 1.6653345369377348E-16 2.8796409701215E-16 8.326672684688674E-  
17 -1.1934897514720433E-15 1.249000902703301E-16 1.0000000000000007 -  
6.661338147750939E-16 2.7755575615628914E-17 -1.8735013540549517E-16 -  
3.608224830031759E-16 -5.134781488891349E-16 -7.771561172376096E-16 9.43689570931383E-  
16 1.249000902703301E-16 4.3021142204224816E-16 -1.0408340855860843E-15  
1.734723475976807E-16 4.163336342344337E-16 -5.551115123125783E-16  
  
1.3877787807814457E-17 -1.249000902703301E-16 2.463307335887066E-16 -  
4.649058915617843E-16 -7.494005416219807E-16 1.249000902703301E-16 6.38378239159465E-  
16 0.99999999999999992 -3.0531133177191805E-16 -2.2898349882893854E-16 -  
4.163336342344337E-16 -8.881784197001252E-16 -6.661338147750939E-16  
2.636779683484747E-16 -3.191891195797325E-16 2.2898349882893854E-16 -  
2.706168622523819E-16 0.0 2.3592239273284576E-16 -1.0824674490095276E-15  
  
5.412337245047638E-16 6.38378239159465E-16 -1.457167719820518E-16 5.551115123125783E-  
16 1.6653345369377348E-15 4.440892098500626E-16 -1.3877787807814457E-16 -  
1.887379141862766E-15 0.9999999999999998 -5.551115123125783E-17 2.220446049250313E-16  
8.326672684688674E-17 1.1657341758564144E-15 -7.771561172376096E-16 -  
5.551115123125783E-17 -3.0531133177191805E-16 8.604228440844963E-16 -  
7.632783294297951E-17 -8.326672684688674E-16 8.326672684688674E-16

2.4980018054066022E-15 -1.970645868709653E-15 -6.498274141009119E-15 -  
1.915134717478395E-15 -5.88418203051333E-15 -2.67841304690819E-15 5.717648576819556E-  
15 3.552713678800501E-15 -1.1102230246251565E-16 1.0000000000000016 -  
2.0816681711721685E-15 -2.1371793224034263E-15 -1.9984014443252818E-15 -  
5.551115123125783E-17 -1.5265566588595902E-16 1.3183898417423734E-15 -  
4.3021142204224816E-16 -2.4077961846558082E-15 2.831068712794149E-15 -  
3.2751579226442118E-15  
  
-1.4849232954361469E-15 -6.661338147750939E-16 9.922618282587337E-16 -  
8.326672684688674E-17 -1.1657341758564144E-15 0.0 -2.7755575615628914E-17 -  
1.1102230246251565E-16 -3.3306690738754696E-16 3.3306690738754696E-16  
0.9999999999999998 0.0 -9.992007221626409E-16 1.304512053934559E-15 -  
1.942890293094024E-16 4.718447854656915E-16 -1.1657341758564144E-15  
5.273559366969494E-16 4.440892098500626E-16 2.7755575615628914E-16  
  
1.4992347641129555E-15 1.3027773304585821E-15 -5.692061405548898E-16  
5.065392549852277E-16 1.904726376622534E-15 -6.140921104957897E-16 -  
9.610368056911511E-16 1.412064909445121E-15 1.0321604682062002E-15  
1.6089560239684886E-16 3.5561831257524545E-16 1.0000000000000009 2.256875242245826E-15  
-9.280770596475918E-16 6.539907504432563E-16 -7.884318198314588E-16  
6.652664530371055E-16 -2.5652223401007035E-16 1.734723475976807E-17  
1.438085761584773E-15  
  
-1.7763568394002505E-15 -4.996003610813204E-16 -8.153200337090993E-16 -  
9.159339953157541E-16 -4.052314039881821E-15 -1.5265566588595902E-15  
3.913536161803677E-15 3.4416913763379853E-15 5.551115123125783E-16 6.938893903907228E-  
16 -3.3306690738754696E-16 1.4710455076283324E-15 0.9999999999999978  
7.494005416219807E-16 9.43689570931383E-16 2.914335439641036E-16 -2.1371793224034263E-  
15 1.457167719820518E-16 1.3877787807814457E-15 -2.1649348980190553E-15  
  
3.5388358909926865E-16 -2.2343238370581275E-15 -8.083811398051921E-16 -  
3.469446951953614E-16 -4.6074255521944E-15 -7.632783294297951E-16 4.066191827689636E-  
15 -2.220446049250313E-16 -1.5265566588595902E-15 1.5890067039947553E-15 -  
2.6645352591003757E-15 -2.3592239273284576E-15 -3.83026943495679E-15  
1.00000000000000036 -3.677613769070831E-15 1.5057399771478686E-15 -1.7069679003611782E-  
15 -1.4606371667724716E-15 5.551115123125783E-17 -2.7200464103316335E-15  
  
2.636779683484747E-16 0.0 -1.0061396160665481E-15 -7.494005416219807E-16 -  
9.992007221626409E-16 -3.0808688933348094E-15 1.1657341758564144E-15 -  
1.2212453270876722E-15 7.216449660063518E-16 8.743006318923108E-16 -  
1.4988010832439613E-15 1.0547118733938987E-15 -1.942890293094024E-15  
1.7208456881689926E-15 1.0000000000000004 -8.049116928532385E-16 -7.216449660063518E-  
16 1.734723475976807E-16 1.6653345369377348E-15 -1.1102230246251565E-16  
  
-6.938893903907228E-16 2.525757381022231E-15 1.887379141862766E-15 2.581268532253489E-  
15 7.216449660063518E-15 -5.551115123125783E-17 -3.552713678800501E-15 -  
1.1102230246251565E-16 1.1657341758564144E-15 -9.575673587391975E-16  
9.43689570931383E-16 1.7763568394002505E-15 2.831068712794149E-15 -  
1.1657341758564144E-15 1.5265566588595902E-15 0.9999999999999979 2.525757381022231E-15  
1.762479051592436E-15 -8.326672684688674E-16 5.162537064506978E-15  
  
-1.4016565685892601E-15 -1.4710455076283324E-15 -8.951173136040325E-16 -  
1.9984014443252818E-15 -4.884981308350689E-15 -1.2490009027033011E-15  
1.4988010832439613E-15 1.6653345369377348E-15 2.220446049250313E-16 6.38378239159465E-  
16 -6.106226635438361E-16 1.3877787807814457E-16 -1.9984014443252818E-15  
8.881784197001252E-16 1.1102230246251565E-15 8.326672684688674E-16 0.999999999999998  
4.926614671774132E-16 1.7208456881689926E-15 -1.4988010832439613E-15  
  
-9.43689570931383E-16 -9.298117831235686E-16 2.976785484776201E-15 -  
1.942890293094024E-16 -1.8318679906315083E-15 6.938893903907228E-16 -  
5.828670879282072E-16 1.27675647831893E-15 -8.049116928532385E-16 -6.175615574477433E-  
16 6.800116025829084E-16 -4.3021142204224816E-16 -1.2351231148954867E-15  
9.298117831235686E-16 -3.608224830031759E-16 1.3183898417423734E-15 -  
2.3592239273284576E-15 0.9999999999999997 1.3877787807814457E-15 -2.7755575615628914E-  
16  
  
-1.2628786905111156E-15 4.440892098500626E-16 -7.615436059538183E-16 -  
7.494005416219807E-16 -2.3869795029440866E-15 -1.4155343563970746E-15

```

1.609823385706477E-15 6.050715484207103E-15 1.4432899320127035E-15
1.1102230246251565E-15 -8.326672684688674E-16 2.5396351688300456E-15 -
6.800116025829084E-16 9.71445146547012E-16 2.761679773755077E-15 -1.457167719820518E-
16 -1.3600232051658168E-15 1.033895191682177E-15 1.0000000000000013 -
5.551115123125783E-17

-6.765421556309548E-16 -4.371503159461554E-16 1.726049858596923E-16 -
3.400058012914542E-16 -1.3877787807814457E-16 2.9004576518332215E-15 -
3.885780586188048E-15 2.400857290751901E-15 1.6653345369377348E-16 -
1.186550857568136E-15 2.310651670001107E-15 9.159339953157541E-16 9.020562075079397E-
16 -1.7139067942650854E-15 2.0122792321330962E-16 8.569533971325427E-16 -
6.453171330633722E-16 -3.712308238590367E-16 1.5959455978986625E-16 0.9999999999999989

```

**Мера обусловленности:** 16179.749612466816

## Вывод

Мы рассмотрели решение различных систем линейных алгебраических уравнений методом Гаусса. Можно сделать следующие выводы:

- Если в процессе прямого хода метода Гаусса одно или несколько уравнений принимают вид  $f = 0$ , где  $f$  - некоторое число, отличное от нуля, то система несовместна.
- Если в конце прямого хода метода Гаусса мы получаем систему, число уравнений в которой совпадает с числом неизвестных переменных, то система совместна и определена, то есть, имеет единственное решение, которое определяется при проведении обратного хода метода Гаусса.
- Если после завершения прямого хода метода Гаусса в полученной СЛАУ число уравнений меньше числа неизвестных переменных, то система совместна и имеет бесконечное множество решений, которые находятся при обратном ходе метода Гаусса.