

# НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО

Факультет программной инженерии и компьютерной техники

Направление подготовки 09.03.04 - Программная инженерия

Дисциплина – Вычислительная математика

## Лабораторная работа №1 Вариант №11

Выполнил: Мухсинов С.П

Группа: Р3217

Преподаватель: Малышева  
Т.А

Санкт-Петербург

2024

## Цель работы:

Научиться решать СЛАУ при помощи численных методов и написать программу, которая находит приближенное решение, получая на вход матрицу из файла или консоли.

## Задание:

1. № варианта определяется как номер в списке группы согласно ИСУ.
2. В программе численный метод должен быть реализован в виде отдельной подпрограммы/метода/класса, в который исходные/выходные данные передаются в качестве параметров.
3. Размерность матрицы  $n \leq 20$  (задается из файла или с клавиатуры – по выбору конечного пользователя).
4. Должна быть реализована возможность ввода коэффициентов матрицы, как с клавиатуры, так и из файла (по выбору конечного пользователя).

### 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, расчетные формулы:

Метод Гаусса с выбором главного элемента - является одной из модификаций метода Гаусса. Идеей метода Гаусса с выбором главного элемента является такая перестановка уравнений, чтобы на  $k$ -ом шаге исключения ведущим элементом  $a_{ii}$  оказывался наибольший по модулю элемент  $k$ -го столбца.

То есть на очередном шаге  $k$  в уравнениях, начиная от  $k$  до последнего ( $i = k, k + 1, \dots, n$ ) в столбце  $k$  выбирают максимальный по модулю элемент и строки  $i$  и  $k$  меняются местами. Это выбор главного элемента «по столбцу».

Данный метод применяется, чтобы избежать вычислительной погрешности при значениях ведущих элементов, близких к нулю по абсолютной величине,

## Блок-схема метода Гаусса

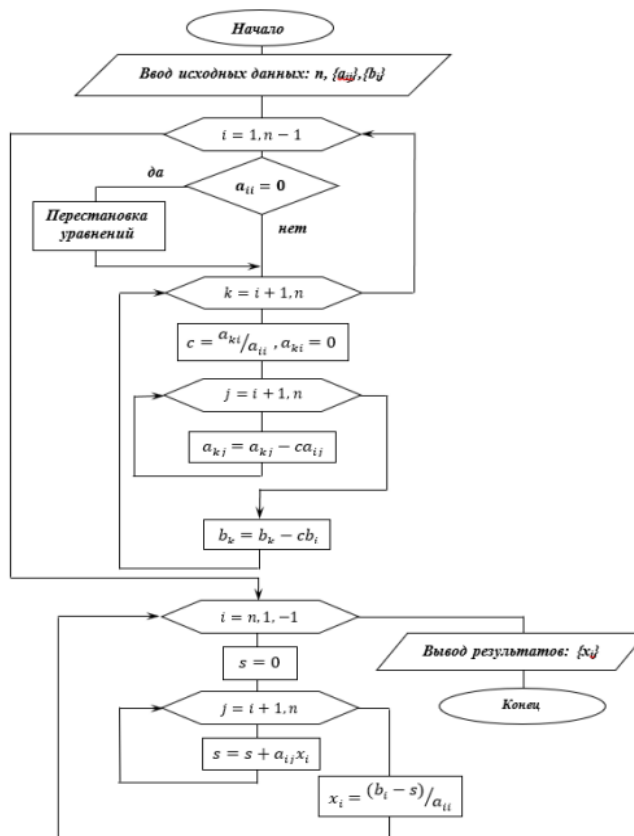
Первый цикл с переменной цикла  $i$  реализует прямой ход, а второй – обратный ход метода.

$i$  – номер неизвестного, которое исключается из оставшихся  $n - 1$

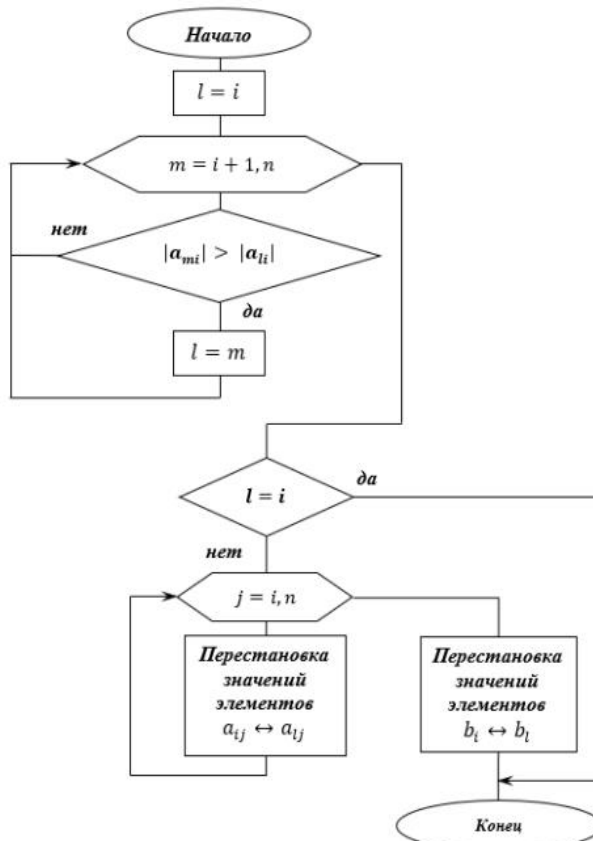
уравнений при прямом ходе (а также номер уравнения, из которого исключается  $x_i$ ) и номер неизвестного, которое определяется из  $i$ -го уравнения при обратном ходе;

$k$  – номер уравнения, из которого исключается неизвестное  $x_i$  при прямом ходе;

$j$  – номер столбца при прямом ходе и номер уже найденного неизвестного при обратном ходе.



## Блок-схема метод Гаусса с выбором главного элемента



$l$  – номер наибольшего по абсолютной величине элемента матрицы в столбце с номером  $i$  (т. е. среди элементов  $a_{li}, \dots, a_{mi}, \dots, a_{ni}$ );

$m$  – текущий номер элемента, с которым происходит сравнение;

Выбор главного элемента осуществляется по столбцу

## Листинг программы:

```
package main;

public class Solver {
    float[][] A;

    float[][] enteredA;
    float[] enteredB;
    float[] B;

    float[] X;

    float[] R;
    final int N;
    float D;

    public Solver(int N, float[][] A, float[] B){
        this.N = N;
        this.A = A;
        this.B = B;
        this.enteredA = new float[N][N];
        this.enteredB = new float[N];
        for(int i = 0; i < N; i++){
            System.arraycopy(A[i], 0, enteredA[i], 0, N);
        }
        System.arraycopy(B, 0, enteredB, 0, N);
        X = new float[N];
        R = new float[N];
    }

    public void solve(){
        int p = forwardMove();
        calculateDeterminant(p);
        System.out.println("Определитель матрицы коэффициентов: " + D);
        reversalMove();
        calculateResidualVector();
        System.out.println();
        System.out.println();
        System.out.println("Треугольная матрица и преобразованный столбец B:");

        printMatrix();
        System.out.println();
        printAnswer();
        System.out.println();
        printResidualVector();
    }

    private int chooseMainElement(int k){
        float mainE = A[k][k];
        int mainRow = k;
        for(int i = k+1; i < N; i++){
            if(Math.abs(A[i][k]) > Math.abs(mainE)){
                mainE = A[i][k];
                mainRow = i;
            }
        }
        if(mainRow != k){
            for(int i = k; i < N; i++){
                float a = A[k][i];
                A[k][i] = A[mainRow][i];
                A[mainRow][i] = a;
            }
        }
    }
}
```

```

        float a = B[k];
        B[k] = B[mainRow];
        B[mainRow] = a;
        return 1;
    }
    return 0;
}

private int forwardMove(){
    int permutations = 0;
    for(int i = 0; i < N - 1; i++){
        permutations += chooseMainElement(i);
        for(int j = i+1; j < N; j++){
            float c = A[j][i] / A[i][i];
            for(int k = i; k < N; k++){
                A[j][k] = A[j][k] - c * A[i][k];
            }
            B[j] = B[j] - c * B[i];
        }
    }
    return permutations;
}

private void calculateDeterminant(int permutations){
    D = 1;
    for(int i = 0; i < N; i++){
        D *= A[i][i];
    }
    D *= Math.pow(-1, permutations);
}

private void reversalMove(){
    X[N-1] = B[N-1]/A[N-1][N-1];
    for(int i = N-2; i >= 0; i--){
        float s = 0;
        for(int j = i + 1; j < N; j++){
            s += A[i][j]*X[j];
        }
        X[i] = (B[i] - s)/A[i][i];
    }
}

private void calculateResidualVector(){
    for (int i = 0; i < N; i++) {
        float s = 0;
        for (int j = 0; j < N; j++) {
            s += enteredA[i][j] * X[j];
        }
        R[i] = s - enteredB[i];
    }
}

private void printResidualVector(){
    System.out.println("Вектор невязок: ");
    for (int i = 0; i < N; i++) {
        System.out.print(R[i] + "\t");
    }
}
}

```

## Примеры работы программы:

### Пример 1:

Вы хотите ввести данные с файла или с клавиатуры?

1) С файла

2) С клавиатуры

2

Введите размерность матрицы:

3

Введите коэффициенты 1-го уравнения:

10 -7 0

Введите значение правой части 1-го уравнения:

7

Введите коэффициенты 2-го уравнения:

-3 2 6

Введите значение правой части 2-го уравнения:

4

Введите коэффициенты 3-го уравнения:

5 -1 5

Введите значение правой части 3-го уравнения:

6

Определитель матрицы коэффициентов: -155.0

Треугольная матрица и преобразованный столбец В:

10.0	-7.0	0.0	7.0
0.0	2.5	5.0	2.5
0.0	0.0	6.2000003	6.2000001

$x_1 = -1.9073487E-7$

$x_2 = -1.0000002$

$x_3 = 1.0000001$

Вектор невязок:

0.0 9.536743E-7 0.0

### Пример 2:

Вы хотите ввести данные с файла или с клавиатуры?

1) С файла

2) С клавиатуры

1

Введите путь к файлу:

input

Прочитанная матрица коэффициентов и правые части уравнений:

0.0	1.0	2.0	3.0
4.0	5.0	6.0	10.0
17.0	8.0	9.0	1.0

Определитель матрицы коэффициентов: -40.000008

Треугольная матрица и преобразованный столбец В:

17.0	8.0	9.0	1.0
0.0	3.1176472	3.8823528	9.764706
0.0	0.0	0.7547171	-0.13207531

$x_1 = -1.425$

$x_2 = 3.3499994$

$x_3 = -0.17499976$

Вектор невязок:

0.0   -1.9073486E-6   -8.34465E-7

## Вывод:

В ходе выполнения лабораторной работы мы познакомились с численными методами решения математических задач и реализовали программу, находящую приближенные решения СЛАУ при помощи метода Гаусса с выбором главного элемента по столбцам.