МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего образования
«Дальневосточный федеральный университет»
(ДВФУ)

ИНСТИТУТ МАТЕМАТИКИ И КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ (ШКОЛА)

Департамент математического и компьютерного моделирования

ОТЧЕТ

к лабораторной работе №13 по дисциплине
«Вычислительная математика»

Направление подготовки
01.03.02 «Прикладная математика и информатика»

Выполнил студент гр.
Б9121-01.03.02сп1
Рютин Д. С. __________
(Ф.И.О.) *(подпись)*

« 09 » января 20 24 г.

г. Владивосток

2024

Содержание

Введение

Цель данной лабораторной работы – запрограммировать и протестировать метод Гаусса с выбором главного элемента для нахождения решения СЛАУ[1].

[1]Систем линейных алгебраических уравнений.

Описание алгоритма

Рассмотрим расширенную прямоугольную матрицу состоящую из коэффициентов:

$$M = \left(\begin{array}{cccccc|c} a_{11} & a_{12} & \cdots & a_{1q} & \cdots & a_{1n} & b_1 \\ a_{21} & a_{22} & \cdots & a_{2q} & \cdots & a_{2n} & b_2 \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \vdots \\ a_{p1} & a_{p2} & \cdots & a_{pq} & \cdots & a_{pn} & b_p \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nq} & \cdots & a_{nn} & b_n \end{array}\right). \tag{1}$$

Выберем наибольший по модулю элемент a_{pq}, не принадлежащий столбцу свободных членов матрицы M. Этот элемент называется *главным элементом*. Строка и столбец матрицы M, содержащие главный элемент, называются *главной строкой* и *главным столбцом* соответственно.

Пусть $i = \overline{1,n},\ i \neq p$, из каждой i-й строки матрицы M вычтем p-ю строку умноженную на $-\ a_{iq}/a_{jq}$. Очевидно, что после этого в q-столбце все элементы будут равны нулю кроме элемента a_{pq}. Назовем полученную матрицу M_*. Скажем, что $M^{(1)}$ – матрица M_* после удаления из нее главной строки и главного столбца.

Проделываем эти же действия с матрицей $M^{(1)}$, получаем матрицу $M^{(2)}$. Повторяем алгоритм, пока не получим матрицу $M^{(n-1)}$, которая из себя представляет двух элементную матрицу – строку, которая также является главной.

Для получения системы с треугольной матрицей, эквивалентной изначальной системе, объединяем все главные строки матриц $M,\ M^{(1)},\ M^{(2)},\ \ldots,$ $M^{(n-1)}$, начиная с последней $M^{(n-1)}$.

Решить получившуюся систему можно, последовательно идя по системе находя значения новых неизвестных, как примере с следующей треугольной системой:

$$\begin{cases} x_1 = \beta_1; \\ x_2 + \alpha_{21}x_1 = \beta_2; \\ x_3 + \alpha_{32}x_2 + \alpha_{31}x_1 = \beta_3; \\ x_4 + \alpha_{43}x_3 + \alpha_{42}x_2 + \alpha_{41}x_1 = \beta_4. \end{cases}$$

Решение:

$$\begin{aligned} x_1 &= \beta_1, \\ x_2 &= \beta_2 - \alpha_{21}x_1, \\ x_3 &= \beta_3 - \alpha_{32}x_2 - \alpha_{31}x_1, \\ x_4 &= \beta_4 - \alpha_{43}x_3 - \alpha_{42}x_2 - \alpha_{41}x_1. \end{aligned}$$

Тестирование

Пример 1

$$A = \begin{pmatrix} 0.411 & 0.421 & -0.333 & 0.313 & -0.141 & -0.381 & 0.245 \\ 0.241 & 0.705 & 0.139 & -0.409 & 0.321 & 0.062 & 0.101 \\ 0.123 & -0.239 & 0.502 & 0.901 & 0.243 & 0.819 & 0.321 \\ 0.413 & 0.309 & 0.801 & 0.865 & 0.423 & 0.118 & 0.183 \\ 0.241 & -0.221 & -0.243 & 0.134 & 1.274 & 0.712 & 0.423 \\ 0.281 & 0.525 & 0.719 & 0.118 & -0.974 & 0.808 & 0.923 \\ 0.246 & -0.301 & 0.231 & 0.813 & -0.702 & 1.223 & 1.105 \end{pmatrix}, \overline{b} = \begin{pmatrix} 0.096 \\ 1.252 \\ 1.024 \\ 1.023 \\ 1.155 \\ 1.937 \\ 1.673 \end{pmatrix}.$$

Результат и погрешность:

$$\bar{x} = \begin{pmatrix} 11.092 & -2.516 & 0.721 & -2.545 & -1.605 & 3.624 & -4.95 \end{pmatrix}$$

$$\text{error} = \begin{pmatrix} 0.11 & -0.23 & -0.68 & -0.23 & -0.34 & -0.34 & -0.45 \end{pmatrix} \cdot 1.97 \cdot 10^{-15}$$

Пример 2

$$A = \begin{pmatrix} 2.2 & 4 & -3 & 1.5 & 0.6 & 2 & 0.7 \\ 4 & 3.2 & 1.5 & -0.7 & -0.8 & 3 & 1 \\ -3 & 3.2 & 1.5 & -0.7 & -0.8 & 3 & 1 \\ 1.5 & -0.7 & 0.9 & 2.2 & 4 & 3 & 1 \\ 0.6 & -0.8 & 3 & 4 & 3.2 & 0.6 & 0.7 \\ 2 & 3 & 2 & 3 & 0.6 & 2.2 & 4 \\ 0.7 & 1 & 2 & 1 & 0.7 & 4 & 3.2 \end{pmatrix}, \overline{b} = \begin{pmatrix} 3.2 \\ 4.3 \\ -0.1 \\ 3.5 \\ 5.3 \\ 9 \\ 3.7 \end{pmatrix}.$$

Результат и погрешность:

$$\bar{x} = \begin{pmatrix} 0.629 & 0.959 & 0.629 & 0.124 & 0.944 & -0.838 & 1.128 \end{pmatrix}$$

$$\text{error} = \begin{pmatrix} -0.182 & -0.73 & -0.034 & 0 & -0.365 & 0 & -0.547 \end{pmatrix} \cdot 2.434 \cdot 10^{-15}$$

Вывод

Алгоритм выдает довольно точный ответ.

Заключение

В этой лабораторной работе была проведена работа по программированию и тестированию метода Гаусса с выбором главного элемента для решения системы линейных алгебраических уравнений.

Приложение

Код 1: main.cpp

```
#include <types.h>
#include <options.h>
#include <latex.h>
#include <gaus.h>

#include <iostream>
#include <string>

int main(int argc, char* argv[]) {

    int PRECISION = 4;

    // Options processing

    op::OpController ops;
    bool latex = false;

    ops.addParser({"-l", "--latex"}, [&latex](op::OptionState st){
        latex = true;
    });

    ops.addParser({"-p", "--precision"}, [&PRECISION](op::OptionState st){
        int value = -1;
        for (int i = st.pos + 1; i < st.args.size(); i++) {
            if (st.argstate[i] == 0) {
                value = std::stoi(st.args[i]);
                st.argstate[i] = 1;
                break;
            }
        }
        if (value == -1)
            throw op::OpController::Exception{ "Not enough arguments" };

        ltx::PRECISION = value;
        PRECISION = value;
    });

    try {
        ops.ProcessArgs(argc, argv);
    } catch (op::OpController::Exception e) {
        std::cout << "OptionError: " << e.msg << std::endl;
        throw;
    }

    // Data input

```

```
    int n;
    std::cin >> n;

    Mtx A(n, n);
    Vec b(n);
    std::cin >> A >> b;

    // Processing

    auto [exp, res] = my::Gaus(A, b);

    // Result output

    if (latex) {
        std::cout << ltx::latex(A) << " \\cdot \\bar x = " << ltx::latex(b) << std::endl;
        std::cout << "\\bar x = " << ltx::latex(res.transpose()) << std::endl;
        std::cout << "\\operatorname{error} = " << ltx::latex((A * res - b).normalized().transpose())
                  << " \\cdot " << ltx::latex((A * res - b).norm()) << std::endl;
    } else {
        std::cout << "Roots : \n" << res.transpose() << std::endl;
        std::cout << "Error : \n(" << (A * res - b).normalized().transpose() << ") * " << (A * res - b).norm()
    }

}
```

Код 2: gaus.h

```
#ifndef _GAUS_H_
#define _GAUS_H_

#include <types.h>

#include <tuple>
#include <set>
#include <vector>

namespace my {

std::tuple<Mtx, Vec> Gaus(Mtx A, Vec b) {
    std::set<int> rows;
    for (int i = 0; i < A.rows(); i++)
        rows.insert(i);

    Vec res(A.rows());
    std::vector<int> mapping(A.rows());

    while (!rows.empty()) {
        int mx = 0, my = *rows.begin();

        for (auto y : rows)
            for (int x = 0; x < A.cols(); x++)
```

```
                if (std::abs(A(y, x)) > std::abs(A(my, mx))) {
                    my = y;
                    mx = x;
                }


        for (int y = 0; y < A.rows(); y++) {
            if (y == my) continue;

            double coef = A(y, mx) / A(my, mx);
            b(y) -= coef * b(my);
            for (int x = 0; x < A.cols(); x++)
                A(y, x) -= coef * A(my, x);
        }

        res(mx) = A(my, mx);
        mapping[mx] = my;
        rows.erase(my);
    }

    for (int i = 0; i < res.rows(); i++)
        res(i) = b(mapping[i]) / res(i);

    return {A, res};
}

} // namespace my


#endif // _GAUS_H_
```