Федеральное государственное автономное образовательное учреждение высшего

образования «Национальный исследовательский университет ИТМО»

Факультет Программной Инженерии и Компьютерной Техники

Лабораторная работа №1

по дисциплине «Вычислительная математика»

Вариант: 12

Преподаватель:

Малышева Татьяна Алексеевна

Выполнил:

Рубцов Арсений Дмитриевич

Группа: P3206

Санкт — Петербург, 2025

# Цель работы

Изучить численные методы решения систем линейных алгебраических уравнений и реализовать один из них средствами программирования.

# Описания метода

Итерационные методы дают возможность для системы (1) построить последовательность

векторов 𝑥 (0) , 𝑥 (1) , … , 𝑥 (𝑘) , пределом которой должно быть точное решение

𝑥 (∗) : 𝑥 (∗) = lim 𝑘→∞ 𝑥 (𝑘) Построение последовательности заканчивается, как только достигается желаемая точность.

# Листинг программы

<https://github.com/arseeenyyy/computational-mathematics/tree/main/lab1>

#include "GaussSeidel.h"

#include <cmath>

#include <vector>

#include <iostream>

#include <algorithm>

#include <iomanip>

using namespace std;

void GaussSeidel::gaussSeidelMethod(vector<vector<double>> &matrix, vector<double> &x, double accuracy) {

int n = matrix.size();

int iteration = 0;

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

x[i] = matrix[i][n] / matrix[i][i];

}

cout << setw(7) << "№";

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

cout << setw(10) << "x" + to\_string(i + 1);

}

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

cout << setw(1) << "ε" + to\_string(i + 1);

}

cout << endl;

while (true) {

vector<double> new\_x = x;

vector<double> epsilon(n, 0);

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

double sum = 0;

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

if (i != j) sum += matrix[i][j] \* new\_x[j];

}

new\_x[i] = (matrix[i][n] - sum) / matrix[i][i];

epsilon[i] = abs(new\_x[i] - x[i]);

}

cout << setw(7) << iteration + 1;

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

cout << setw(10) << fixed << setprecision(4) << new\_x[i];

}

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

cout << setw(10) << fixed << setprecision(4) << epsilon[i];

}

cout << endl;

double max\_error = \*max\_element(epsilon.begin(), epsilon.end());

if (max\_error <= accuracy) {

break;

}

x = new\_x;

iteration++;

}

cout << "\nMethod converged in " << iteration + 1 << " iterations.\n";

}

# Пример работы программы

format of enter params:

n (int size; 0 <= n <= 20)

accuracy (double)

matrix params:

a\_11 a\_12 ... b\_1n

a\_21 a\_22 ... b\_2n

a\_n1 a\_n2 ... b\_nn

How will you enter matrix dimension? (1 - from the keyboard, 2 - from the file): 2

Enter file name: matrix.txt

Matrix is diagonally dominant

Matrix norm: 9

№ x1 x2 x3 ε1 ε2 ε3

1 -1.2000 -4.3000 1.6200 2.2000 0.7000 0.3800

2 -0.8320 -4.3980 1.4532 0.3680 0.0980 0.1668

3 -0.7515 -4.4613 1.4084 0.0805 0.0633 0.0448

4 -0.7373 -4.4801 1.3989 0.0143 0.0189 0.0095

5 -0.7354 -4.4844 1.3973 0.0019 0.0043 0.0016

Method converged in 5 iterations.

Solution vector:

[x1] = -0.7354

[x2] = -4.4844

[x3] = 1.3973

ε vector:

[ε1] = 0.0019

[ε2] = 0.0043

[ε3] = 0.0016

# Вывод

В результате выполнения лабораторной работы я познакомился с численными методами решения математических задач на примере СЛАУ, реализовав на ЯП C++ метод Гаусса — Зейделя.