

Санкт-Петербургский государственный университет
математико-механический факультет

Программа для решения
системы линейных
алгебраических уравнений
методом Крамера

Бода Александр Павлович

Механика и математическое моделирование 23.Б12-ММ

Рецензент: ассистент кафедры теоретической механики, Нестерчук Г. А.

Санкт-Петербург 2024

Содержание

1	Введение	2
2	Выбор инструментов	2
2.1	Среда разработки	2
2.2	Библиотеки	2
3	Документация по коду	2
3.1	Заголовочный файл <code>cramer.h</code>	2
3.2	Файл реализации <code>cramer.cpp</code>	3
3.2.1	Функция <code>determinant</code>	3
3.2.2	Функция <code>initializeMatrix</code>	4
3.2.3	Функция <code>initializeVector</code>	4
3.2.4	Функция <code>replaceColumn</code>	4
3.2.5	Функция <code>outputSystemLaTeX</code>	5
3.2.6	Функция <code>outputSolutionStepsLaTeX</code>	5
3.2.7	Функция <code>solveSystem</code>	6
3.3	<code>main.cpp</code>	7
4	Примеры решения задач	9
5	Заключение	12

1 Введение

Задача решения системы линейных алгебраических уравнений (СЛАУ) является одной из наиболее распространенных задач в математике и её приложениях. Различные способы решения СЛАУ изучаются студентами Механико-Математического факультета СПбГУ на первом и втором семестре в курсе высшей алгебры. Изучение линейной алгебры побудило меня на создание универсальной программы, которая решала систему линейных уравнений любого количества переменных. В качестве одного из наиболее частоиспользуемых способ решения был выбран метод Крамера. Программа должна была быть написана на языке C++ и выводить все итерации решения в отдельный файл в формате LaTeX.

2 Выбор инструментов

2.1 Среда разработки

Для реализации проекта была выбрана среда разработки Visual Studio 2022.

2.2 Библиотеки

`include <fstream>` - для вывода данных в файл
`include <iostream>` - для работы с потоком и стандартными функциями
`include <cmath>` - для работы с математическими функциями

3 Документация по коду

3.1 Заголовочный файл `cramer.h`

В данном файле названы функции, которые используются в файле `main.cpp` для решения СЛАУ методом Крамера. Всего 7 функция.

```
// cramer.h
#ifndef CRAMER_H
#define CRAMER_H

#include <fstream>

double determinant(double** matrix, int n);
void initializeMatrix(double** A, int n);
void initializeVector(double* b, int n);
void replaceColumn(double** matrix, double* column, int colIndex, int n);
void outputSystemLaTeX(std::ofstream& outfile, double** A, double* b,
    int n);
void outputSolutionStepsLaTeX(std::ofstream& outfile, double** A,
    double* b, double* solutions, int n);
```

```
void solveSystem(double** A, double* b, double* solutions, int n);

#endif
```

3.2 Файл реализации `cramer.cpp`

В этом файле все 7 функций реализованы

• 3.2.1 Функция `determinant`

Функция типа `double`, которая считает определитель квадратной матрицы с помощью рекурентного разложения по первой строке, отдельно рассмотрен случай матрицы, состоящей из одного элемента. В качестве аргумента функция использует динамическую матрицу (динамический массив из динамических массивов) и целочисленный размер этой матрицы. Функция возвращает определитель данной матрицы, тип возвращаемых данных - `double`

```
double determinant(double** matrix, int n) {
    if (n == 1) {
        return matrix[0][0];
    }
    double det = 0;
    double** submatrix = new double* [n - 1];
    for (int i = 0; i < n - 1; ++i) {
        submatrix[i] = new double[n - 1];
    }
    for (int x = 0; x < n; ++x) {
        int subi = 0;
        for (int i = 1; i < n; ++i) {
            int subj = 0;
            for (int j = 0; j < n; ++j) {
                if (j == x) continue;
                submatrix[subi][subj] = matrix[i][j];
                ++subj;
            }
            ++subi;
        }
        det += (pow(-1, x) * matrix[0][x] * determinant(submatrix,
            n - 1));
    }
    for (int i = 0; i < n - 1; ++i) {
        delete[] submatrix[i];
    }
    delete[] submatrix;
    return det;
}
```

• 3.2.2 Функция initializeMatrix

Функция типа void для инициализации матрицы коэффициентов СЛАУ. В качестве аргументов использует квадратную матрицу и её размер. С помощью вложенного цикла матрица поэлементно заполняется пользователем.

```
void initializeMatrix(double** A, int n)
{
    for (int i = 0; i < n; ++i) {
        A[i] = new double[n];
        for (int j = 0; j < n; ++j) {
            std::cout << "Enter coefficient for x_" << j + 1 << "
                in equation " << i + 1 << ": ";
            std::cin >> A[i][j];
        }
    }
}
```

• 3.2.3 Функция initializeVector

Функция типа void для инициализации вектора свободных членов СЛАУ. В качестве аргументов использует динамический массив и его целочисленный размер. С помощью цикла вектор покоординатно заполняется пользователем.

```
void initializeVector(double* b, int n)
{
    for (int i = 0; i < n; ++i) {
        std::cout << "Enter constant for equation " << i + 1 << ": ";
        std::cin >> b[i];
    }
}
```

• 3.2.4 Функция replaceColumn

Функция типа void, которая меняет выбранный столбец некоторой матрицы на другой. В качестве аргументов использует динамическую матрицу, динамический массив (вектор), размер матрицы и массива, а так же номер столбца, который необходимо заменить. Реализована с помощью цикла и поэлементно изменяет выбранный столбец матрицы.

```
void replaceColumn(double** matrix, double* column, int colIndex,
    int n) {
```

```

    for (int i = 0; i < n; ++i) {
        matrix[i][colIndex] = column[i];
    }
}

```

• 3.2.5 Функция outputSystemLaTeX

Функция типа void для вывода СЛАУ в отдельный файл в формате LaTeX. В качестве аргумента использует файл, динамическую матрицу, динамический массив и целочисленный размер матрицы и массива. С помощью цикла выводит СЛАУ в файл.

```

void outputSystemLaTeX(std::ofstream& outfile, double** A, double*
    b, int n) {
    outfile << "\\begin{equation}\\n\\left\\{\\begin{array}{l}" <<
        std::string(n, 'r') << "\\end{array}\\end{array}\\n";
    for (int i = 0; i < n; ++i) {
        for (int j = 0; j < n; ++j) {
            if (j > 0 && A[i][j] >= 0) outfile << "+";
            outfile << A[i][j] << "x_{j}" << j + 1 << " ";
        }
        outfile << " &= " << b[i] << " \\\\n";
    }
    outfile << "\\end{array}\\right.\\n\\end{equation}\\n";
}

```

3.2.6 Функция outputSolutionStepsLaTeX

Функция выводит в отдельный файл все итерации, которые требуются для решения СЛАУ методом Крамера. Сначала вычисляется детерминант матрицы коэффициентов и обозначается как заглавная буква дельта. После чего вычисляются значения переменных. На каждом этапе отдельно вычисляется детерминант матрицы, которая полностью совпадает с матрицей коэффициентов, за исключением одного столбца, который заменён на вектор свободных членов СЛАУ. Детерминант такой матрицы обозначается как дельта с соответствующим нижним индексом.

```

void outputSolutionStepsLaTeX(std::ofstream& outfile, double** A,
    double* b, double* solutions, int n) {
    double detA = determinant(A, n);
    outfile << "\\begin{equation}\\n\\text{Determinant of } A: \\\n" <<
        "\\Delta = " << detA << "\\n\\end{equation}\\n";

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

```

```

double** Ai = new double* [n];
for (int j = 0; j < n; ++j) {
    Ai[j] = new double[n];
    for (int k = 0; k < n; ++k) {
        Ai[j][k] = A[j][k];
    }
}
replaceColumn(Ai, b, i, n);
double detAi = determinant(Ai, n);
outfile << "\\begin{equation}\\n\\Delta_{" << i + 1 << "} = "
        << detAi << "\\n\\end{equation}\\n";
outfile << "\\begin{equation}\\nx_{" << i + 1 << "} = "
        << "\\frac{\\Delta_{" << i + 1 << "}}{\\Delta} = \\frac{"
        << detAi << "}{" << detA << "} = " << solutions[i] << " "
        << "\\n\\end{equation}\\n";
for (int j = 0; j < n; ++j) {
    delete[] Ai[j];
}
delete[] Ai;
}
}

```

• 3.2.7 Функция solveSystem

Функция типа void, которая решает СЛАУ, используя некоторые вышеперечисленные функции, а именно: determinant, replaceColumn. Программа с помощью цикла находит значения каждый из неизвестных.

```

void solveSystem(double** A, double* b, double* solutions, int n) {
    double detA = determinant(A, n);
    for (int i = 0; i < n; ++i) {
        double** Ai = new double* [n];
        for (int j = 0; j < n; ++j) {
            Ai[j] = new double[n];
            for (int k = 0; k < n; ++k) {
                Ai[j][k] = A[j][k];
            }
        }
        replaceColumn(Ai, b, i, n);
        solutions[i] = determinant(Ai, n) / detA;
        for (int j = 0; j < n; ++j) {
            delete[] Ai[j];
        }
        delete[] Ai;
    }
}

```

3.3 main.cpp

В этом файле реализовано сама программа для решения СЛАУ методом Крамера. Используются функции из файла `cramer.h`. Так же несколько строк программы отведено на стандартные для каждой СЛАУ элементы файла, в который выводится решение. Программа выводит все итерации в формате LaTeX в файл под названием `solution.tex`. Если этот файл не существует, то он создаётся автоматически.

При запуске программы пользователем ему предлагается ввести количество переменных. После чего пользователь может последовательно ввести коэффициенты для уравнений из СЛАУ. Последним этапом является инициализация вектора свободных членов СЛАУ. Далее программа запускает решение, выводит ответ в консоль и в файл `soulution.tex`. Если СЛАУ не может быть решена, об этом сообщается пользователю.

```
// main.cpp
#include <iostream>
#include <fstream>
#include "cramer.h"

int main() {
    int n;
    std::cout << "Enter the number of variables: ";
    std::cin >> n;

    double** A = new double* [n];
    double* b = new double[n];
    double* solutions = new double[n];

    std::cout << "Enter the coefficients of matrix A:" << std::endl;
    initializeMatrix(A, n);

    std::cout << "Enter the constants vector b:" << std::endl;
    initializeVector(b, n);

    std::ofstream outfile("solution.tex");
    outfile <<
        "\\documentclass{article}\\n\\usepackage{amsmath}\\n\\begin{document}\\n";
    outfile << "\\section*{Complete Solution of the System of Linear\n";
        Equations}\\n";
    outputSystemLaTeX(outfile, A, b, n);

    double detA = 0;
```



```

detA = determinant(A, n);

if (detA == 0) {
    outfile << "\\text{The determinant of matrix of coefficients
        equals to zero. The system is unsolvable}\\n";
    outfile << "\\end{document}\\n";
    outfile.close();
    std::cerr << "The system does not have a unique solution." <<
        std::endl;
}

else if (detA != 0)
{
    solveSystem(A, b, solutions, n);

    outfile << "\\section*{Solution using Cramer's Method}\\n";
    outputSolutionStepsLaTeX(outfile, A, b, solutions, n);
    outfile << "\\section*{Answer}\\n";
    outfile << "\\begin{equation}\\n\\left\\{\\begin{array}{r}\\n";
    for (int i = 0; i < n; ++i) {
        outfile << "x_" << i + 1 << "} = " << solutions[i] <<
            "\\\\n";
    }
    outfile << "\\end{array}\\right.\\n\\end{equation}\\n";
    outfile << "\\end{document}\\n";
    outfile.close();

    std::cout << "Solution written to solution.tex" << std::endl;

    std::cout << "Solution:" << std::endl;
    for (int i = 0; i < n; ++i) {
        std::cout << "x_" << i + 1 << " = " << solutions[i] <<
            std::endl;
    }
}

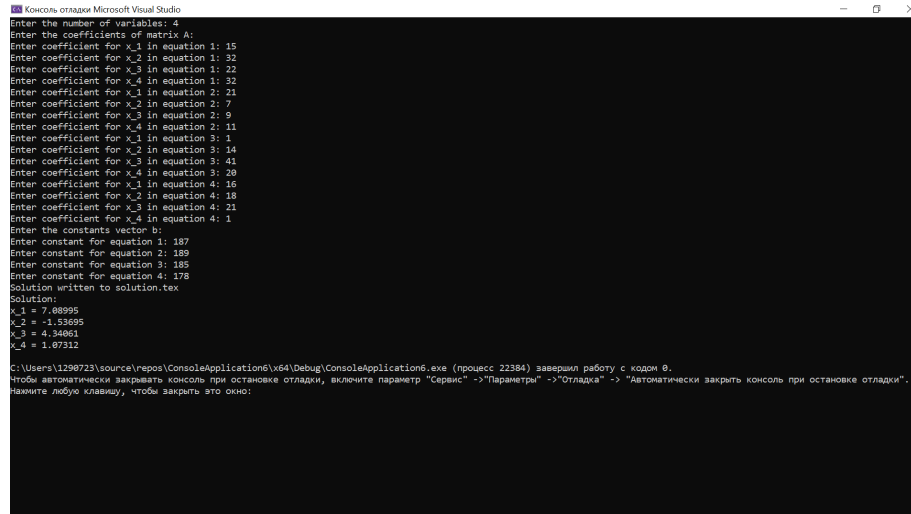
for (int i = 0; i < n; ++i) {
    delete[] A[i];
}
delete[] A;
delete[] b;
delete[] solutions;

return 0;
}

```

4 Примеры решения задач

Рассмотрим пример решения СЛАУ с помощью созданной программы.



```
Консоль отладки Microsoft Visual Studio
Enter the number of variables: 4
Enter the coefficients of matrix A:
Enter coefficient for x_1 in equation 1: 15
Enter coefficient for x_2 in equation 1: 22
Enter coefficient for x_3 in equation 1: 22
Enter coefficient for x_4 in equation 1: 32
Enter coefficient for x_1 in equation 2: 21
Enter coefficient for x_2 in equation 2: 7
Enter coefficient for x_3 in equation 2: 9
Enter coefficient for x_4 in equation 2: 11
Enter coefficient for x_1 in equation 3: 1
Enter coefficient for x_2 in equation 3: 14
Enter coefficient for x_3 in equation 3: 41
Enter coefficient for x_4 in equation 3: 20
Enter coefficient for x_1 in equation 4: 16
Enter coefficient for x_2 in equation 4: 18
Enter coefficient for x_3 in equation 4: 21
Enter coefficient for x_4 in equation 4: 1
Enter the constants vector b:
Enter constant for equation 1: 187
Enter constant for equation 2: 189
Enter constant for equation 3: 185
Enter constant for equation 4: 178
Solution written to solution.tex
Solution:
x_1 = 7.08995
x_2 = -1.53695
x_3 = 4.34861
x_4 = 1.07312
C:\Users\1290723\source\repos\ConsoleApplication6\Debug\ConsoleApplication6.exe (процесс 22384) завершил работу с кодом 0.
Чтобы автоматически закрыть консоль при остановке отладки, включите параметр "Сервис" -> "Параметры" -> "Отладка" -> "Автоматически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно.
```

Рис. 1: Ввод данных в консоль

Далее представлен код в формате LaTeX, который был выведен в файл `solutin.tex` при решении данной СЛАУ.

```
\documentclass{article}
\usepackage{amsmath}
\begin{document}
\section*{Complete Solution of the System of Linear Equations}
\begin{equation}
\left\{\begin{array}{rrrr}
15x_{\{1\}} + 32x_{\{2\}} + 22x_{\{3\}} + 32x_{\{4\}} &=& 187 \\
21x_{\{1\}} + 7x_{\{2\}} + 9x_{\{3\}} + 11x_{\{4\}} &=& 189 \\
1x_{\{1\}} + 14x_{\{2\}} + 41x_{\{3\}} + 20x_{\{4\}} &=& 185 \\
16x_{\{1\}} + 18x_{\{2\}} + 21x_{\{3\}} + 1x_{\{4\}} &=& 178
\end{array}\right.
\end{equation}
\section*{Solution using Cramer's Method}
\begin{equation}
\text{Determinant of } A: \Delta = 374869
\end{equation}
\begin{equation}
\Delta_{\{1\}} = 2.6578e+06
\end{equation}
\begin{equation}
x_{\{1\}} = \frac{\Delta_{\{1\}}}{\Delta} = \frac{2.6578e+06}{374869} = 7.08995
\end{equation}
```

```

\begin{equation}
\Delta_{\{2\}} = -576154
\end{equation}
\begin{equation}
x_{\{2\}} = \frac{\Delta_{\{2\}}}{\Delta} = \frac{-576154}{374869} = -1.53695
\end{equation}
\begin{equation}
\Delta_{\{3\}} = 1.62716\text{e}+06
\end{equation}
\begin{equation}
x_{\{3\}} = \frac{\Delta_{\{3\}}}{\Delta} = \frac{1.62716\text{e}+06}{374869} = 4.34061
\end{equation}
\begin{equation}
\Delta_{\{4\}} = 402278
\end{equation}
\begin{equation}
x_{\{4\}} = \frac{\Delta_{\{4\}}}{\Delta} = \frac{402278}{374869} = 1.07312
\end{equation}
\section*{Answer}
\begin{equation}
\left\{\begin{array}{r}
x_{\{1\}} = 7.08995\\
x_{\{2\}} = -1.53695\\
x_{\{3\}} = 4.34061\\
x_{\{4\}} = 1.07312
\end{array}\right.
\end{equation}
\end{document}

```

Complete Solution of the System of Linear Equations

$$\begin{cases} 15x_1 + 32x_2 + 22x_3 + 32x_4 = 187 \\ 21x_1 + 7x_2 + 9x_3 + 11x_4 = 189 \\ 1x_1 + 14x_2 + 41x_3 + 20x_4 = 185 \\ 16x_1 + 18x_2 + 21x_3 + 1x_4 = 178 \end{cases} \quad (1)$$

Solution using Cramer's Method

$$\text{Determinant of } A : \Delta = 374869 \quad (2)$$

$$\Delta_1 = 2.6578e + 06 \quad (3)$$

$$x_1 = \frac{\Delta_1}{\Delta} = \frac{2.6578e + 06}{374869} = 7.08995 \quad (4)$$

$$\Delta_2 = -576154 \quad (5)$$

$$x_2 = \frac{\Delta_2}{\Delta} = \frac{-576154}{374869} = -1.53695 \quad (6)$$

$$\Delta_3 = 1.62716e + 06 \quad (7)$$

$$x_3 = \frac{\Delta_3}{\Delta} = \frac{1.62716e + 06}{374869} = 4.34061 \quad (8)$$

$$\Delta_4 = 402278 \quad (9)$$

$$x_4 = \frac{\Delta_4}{\Delta} = \frac{402278}{374869} = 1.07312 \quad (10)$$

Answer

$$\begin{cases} x_1 = 7.08995 \\ x_2 = -1.53695 \\ x_3 = 4.34061 \\ x_4 = 1.07312 \end{cases} \quad (11)$$

Рис. 2: Скомпилированный файл solution.tex

5 Заключение

В заключении можно сказать, что проект, целью которого было создать программу для решение СЛАУ методом Крамера, успешно реализован. Созданная программа не имеет видимых замечаний и может быть использована без каких-либо доработок.