

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ
ФЕДЕРАЦИИ**
Федеральное государственное автономное образовательное учреждение
высшего образования
**«Нижегородский государственный университет им. Н.И. Лобачевского»
Национальный исследовательский университет**

**Институт информационных технологий, математики и механики
Кафедра дифференциальных уравнений, математического и численного
анализа**

ОТЧЕТ ПО УЧЕБНОЙ ПРАКТИКЕ
**«Решение систем линейных уравнений различными
методами»**

Выполнил:
студент группы 381706-1
Денисов В.Л.

Проверил:
старший преподаватель
кафедры ДУМЧА, кандидат
физико-математических наук,
Эгамов А.И.

Нижний Новгород
2019

Содержание

1.	Введение	3
2.	Постановка задачи	4
3.	Теоретическая часть	5
3.1	Метод Гаусса	5
3.2	Метод Крамера	6
3.3	Метод простой итерации	7
3.4	Метод Зейделя	8
3.5	Метод верхней релаксации.....	9
3.6	Метод сопряженных градиентов	10
3.7	Метод LU-разложения	11
4.	Сравнение результатов.....	13
5.	Пример работы программы	14
6.	Заключение.....	16
7.	Литература	17

1. Введение

Лабораторная работа направлена на изучение различных способов решения систем линейных уравнений.

Существуют различные способы решения СЛАУ. Причем разделить их можно на прямые и итерационные.

К прямым методам относятся такие методы, которые в предположении, что вычисления ведутся без округлений, позволяют получить точные значения неизвестных. Они просты, универсальны и используются для широкого класса систем. Однако они не применимы к системам больших порядков ($n < 200$) и к плохо обусловленным системам из-за возникновения больших погрешностей.

К итерационным относятся методы, которые даже в предположении, что вычисления ведутся без округлений, позволяют получить решение системы лишь с заданной точностью.

В данной лабораторной работе будут рассмотрены ряд прямых методов: Гаусса, Крамера; LU-разложения, а также несколько итерационных: Зейделя, верхней релаксации, простой итерации, сопряженных градиентов. Подробное описание каждого из этих методов будет дано в теоретической части.

Введем ряд понятий, которые потребуются для изучения способов решения СЛАУ.

Матрица – математический объект, записываемый в виде прямоугольной таблицы элементов (например, целых или действительных чисел), которая представляет собой совокупность строк и столбцов, на пересечении которых находятся её элементы.

Матрица A является **симметричной**, если она совпадает со своей транспонированной, т.е. $A = A^T$.

Симметричная матрица A называется **положительно определенной**, если для любого вектора x выполняется следующее неравенство: $x^T A x > 0$.

Квадратная матрица A обладает свойством **диагонального преобладания**, если выполняется условие

$$|a_{ii}| \geq \sum_{j \neq i} |a_{ij}|, \text{ где } i = 1, 2, \dots, n$$

2. Постановка задачи

В рамках лабораторной работы ставится задача реализации программного комплекса, который позволит решать системы линейных алгебраических уравнений различными способами.

Ограничимся тем, что порядок решаемой системы находится в границах от 2 до 12.

Программный комплекс должен обладать следующими возможностями:

- Получение матрицы коэффициентов СЛАУ и вектора свободных членов в двух режимах: ручной ввод или автоматическая генерация,
- Нахождение решения СЛАУ и вывод полученных результатов на консоль.

В качестве различных методов решения должны быть представлены следующие:

- Метод Гаусса,
- Метод Крамера,
- Метод простой итерации,
- Метод Зейделя,
- Метод верхней релаксации,
- Метод сопряженных градиентов,
- Метод LU-разложения.

Для работы с каждым из них будет представлен отдельный класс. Демонстрация работы вынесена в отдельный модуль.

Программное решение будет выглядеть следующим образом:

1. Модуль `main` – поддержка пользовательского интерфейса для работы с программой.
2. Модуль `IO` – поддержка ввода/вывода матрицы коэффициентов системой линейных уравнений и вектора свободных членов.
3. Модуль `Gauss` – работа со СЛАУ с помощью метода Гаусса
4. Модуль `Kramer` – работа со СЛАУ при помощи метода Крамера.
5. Модуль `Simple_Iter` – работа со СЛАУ при помощи метода простой итерации.
6. Модуль `Zeidel` – работа со СЛАУ при помощи метода Зейделя.
7. Модуль `Relax` – работа со СЛАУ при помощи метода верхней релаксации.
8. Модуль `Gradient` – работа со СЛАУ при помощи метода сопряженных градиентов.
9. Модуль `LU-Decomposition` – работа со СЛАУ при помощи LU-разложения.

3. Теоретическая часть

3.1 Метод Гаусса

Метод Гаусса – классический прямой метод решения СЛАУ. С помощью него последовательно исключаются переменные, и с помощью элементарных преобразований система уравнений приводится к равносильной системе треугольного вида, из которой последовательно, начиная с последних (по номеру), находятся все переменные системы.

Алгоритм решения СЛАУ методом Гаусса подразделяется на два этапа: прямой и обратный ход.

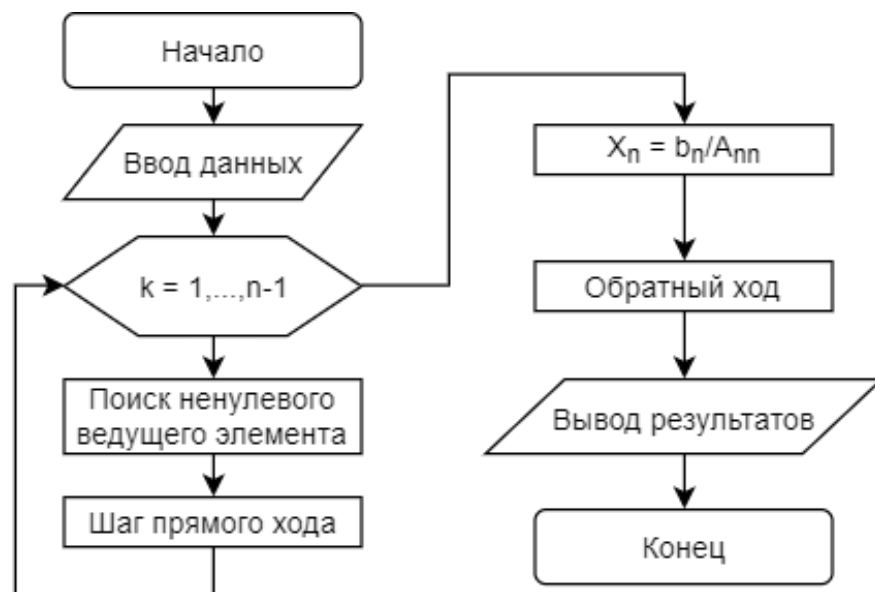


Рисунок 1 Блок-схема метода Гаусса

Прямой ход заключается в том, что путём элементарных преобразований над строками систему приводят к ступенчатому виду. Для этого среди элементов первого столбца матрицы выбирают максимальный по модулю ненулевой элемент, перемещают его в крайнее верхнее положение (происходит выбор ведущего элемента) путем перестановки строк. Затем вычитают получившуюся после перестановки первую строку из остальных, умножив её на величину, равную отношению первого элемента каждой из этих строк к первому элементу первой строки, обнуляя тем самым столбец под ним. После того, как указанные преобразования будут совершены, первую строку и первый столбец мысленно вычёркивают и продолжают пока не останется матрица нулевого размера.

Суть обратного хода заключается в нахождении единственного решения системы линейных уравнений. Эта процедура начинается с последнего уравнения, из которого

выражают соответствующую переменную и подставляют полученное значение в предыдущее уравнения, находя очередную неизвестную. Каждой строчке соответствует ровно одна неизвестная переменная, поэтому на каждом шаге в точности повторяется случай предыдущей строки.

Метод Гаусса имеет сложность $O(n^3)$ при оценке количества действий, требуемых для нахождения решения системы уравнений.

3.2 Метод Крамера

Метод Крамера предназначен для того, чтобы решать системы линейных алгебраических уравнений, в которых число неизвестных переменных равняется числу уравнений, а определитель основной матрицы не равен нулю.



Рисунок 2 Блок-схема метода Крамера

Пусть дана система линейных уравнений $Ax = b$, с n неизвестными где:

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix}$$

Определитель Δ матрицы A отличен от нуля, т.е. $\Delta \neq 0$. Тогда решение системы записывается в виде:

$$x_i = \frac{1}{\Delta} \begin{vmatrix} a_{11} & \cdots & a_{1,i-1} & b_1 & a_{1,i+1} & \cdots & a_{1n} \\ a_{21} & \cdots & a_{2,i-1} & b_2 & a_{2,i+1} & \cdots & a_{2n} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{n-1,1} & \cdots & a_{n-1,i-1} & b_{n-1} & a_{n-1,i+1} & \cdots & a_{n-1,n} \\ a_{n1} & \cdots & a_{n,i-1} & b_n & a_{n,i+1} & \cdots & a_{nn} \end{vmatrix}$$

А именно, i -ый столбец матрицы коэффициентов системы заменяется вектором свободных членов.

Метод Крамера требует вычисления $n + 1$ определителей размерности n . Если использовать для вычисления определителей приведение матрицы к верхне-треугольному виду, то метод имеет сложность по элементарным операциям сложения-умножения порядка $O(n^4)$, что сложнее, чем метод Гаусса при прямом решении системы.

3.3 Метод простой итерации

Альтернативой прямым методам решения СЛАУ являются итерационные методы, основанные на многократном уточнении $x^{(0)}$, заданного приближенного решения системы $Ax = b$. Верхним индексом в скобках здесь и далее по тексту обозначается номер итерации (совокупности повторяющихся действий).

Алгоритм метода простых итераций заключается в выполнении следующих действий.

1. Исходная задача $Ax = b$ преобразуется к равносильному виду:

$$x = \alpha x + \beta$$

где α – квадратная матрица порядка n ; β – столбец. Это преобразование может быть выполнено различными путями, но для обеспечения сходимости итераций нужно добиться выполнения условия $\|\alpha\| < 1$.

2. Столбец β принимается в качестве начального приближения $x^{(0)} = \beta$ и далее многократно выполняются действия по уточнению решения, согласно рекуррентному соотношению:

$$x^{(k+1)} = \alpha x^{(k)} + \beta, \text{ где } k = 0, 1, 2 \dots$$

3. Итерации прерываются при выполнении условия (где $\varepsilon > 0$ – заданная точность, которую необходимо достигнуть при решении задачи) $\|x^{(k+1)} - x^{(k)}\| < \varepsilon$

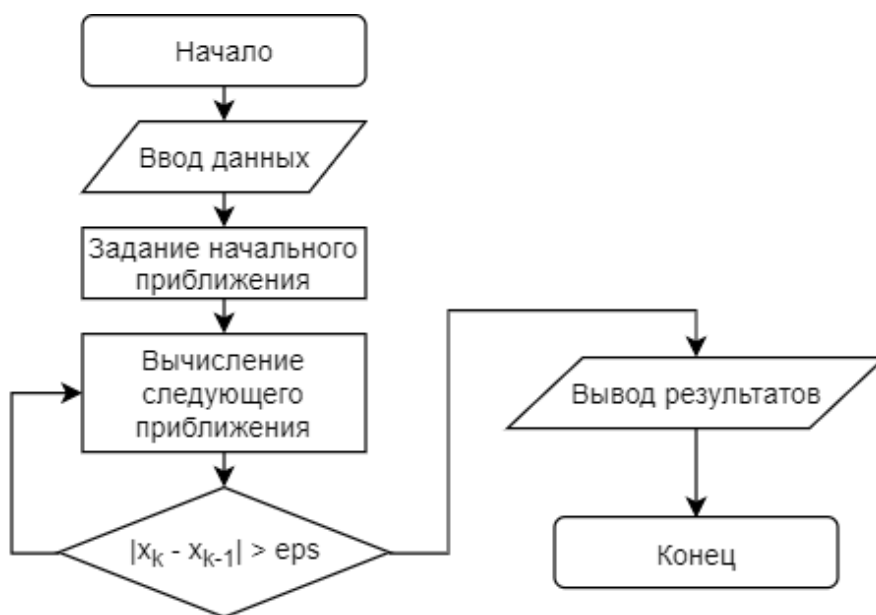


Рисунок 3 Блок-схема метода простых итераций

Начальное приближение $x^{(0)}$ может выбираться произвольно или из некоторых соображений. При этом может использоваться априорная информация о решении или просто "грубая" прикидка.

Кроме того, при рассмотрении метода простых итераций возникает вопрос – в любом ли случае сходится решение?

Существует теорема о достаточном условии сходимости метода простых итераций. Метод простых итераций, реализующийся в процессе последовательных приближений сходится к единственному решению исходной системы $Ax = b$ при любом начальном приближении $x^{(0)}$ со скоростью не медленнее геометрической прогрессии, если какая-либо норма матрицы α меньше единицы, т.е. $\|\alpha\|_s < 1$, где $(s \in \{1, 2, 3\})$.

Однако условие этой теоремы, как достаточное, предъявляет завышенные требования к матрице α , и потому иногда сходимость будет, если даже $\|\alpha\| \geq 1$.

Заметим также, что сходящийся процесс обладает свойством "самоисправляемости", т.е. отдельная ошибка в вычислениях не отразится на окончательном результате, так как ошибочное приближение можно рассматривать, как новое начальное.

Условия сходимости выполняются, если матрица A обладает диагональным преобладанием. Причем чем меньше величина нормы $\|\alpha\|$ этой матрицы, тем быстрее сходимость метода простых итераций.

3.4 Метод Зейделя

Этот метод является модификацией метода простых итераций и в некоторых случаях приводит к более быстрой сходимости.

Итерации по методу Зейделя отличаются от простых итераций тем, что при нахождении i -й компоненты $(k+1)$ -го приближения сразу используются уже найденные компоненты $(k+1)$ -го приближения с меньшими номерами $1, 2, \dots, i-1$.

При рассмотрении развернутой формы системы итерационный процесс записывается в следующем виде:

$$\begin{cases} x_1^{(k+1)} = a_{11}x_1^{(k)} + a_{12}x_2^{(k)} + a_{13}x_3^{(k)} + \dots + a_{1n}x_n^{(k)} + \beta_1 \\ x_2^{(k+1)} = a_{21}x_1^{(k+1)} + a_{22}x_2^{(k)} + a_{23}x_3^{(k)} + \dots + a_{2n}x_n^{(k)} + \beta_2 \\ x_3^{(k+1)} = a_{31}x_1^{(k+1)} + a_{32}x_2^{(k+1)} + a_{33}x_3^{(k)} + \dots + a_{3n}x_n^{(k)} + \beta_3 \\ \dots \\ x_n^{(k+1)} = a_{n1}x_1^{(k+1)} + a_{n2}x_2^{(k+1)} + a_{n3}x_3^{(k+1)} + \dots + a_{nn-1}x_{n-1}^{(k+1)} + a_{nn}x_n^{(k)} + \beta_n \end{cases}$$

В общем виде:

$$x_i^{(k+1)} = \sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} + \sum_{j=i}^n a_{ij}x_j^{(k)} + \beta_i$$

Алгоритм метода Зейделя представляет собой следующий набор действий:

1. Система $Ax = b$ преобразуется к виду $x = \alpha x + \beta$ с преобладанием диагональных элементов в матрице A для обеспечения сходимости метода.
2. Задается произвольное начальное приближение решения $x^{(0)}$ или принимается $x^{(0)} = \beta$, а также задается малое положительное число ε (точность).
3. Выполняются расчеты по рекуррентной формуле, представленной выше для $x^{(k+1)}$.
4. Если выполняется условие окончания $\|x^{(k+1)} - x^{(k)}\| < \varepsilon$, то итерационный процесс завершается и в качестве приближенного решения задачи принимается $x^* \cong x^{(k+1)}$. Иначе $k = k + 1$ и переход к пункту 3.

Как правило, метод Зейделя обеспечивает лучшую сходимость, чем метод простых итераций (за счет накопления информации, полученной при решении предыдущих уравнений). Метод Зейделя может сходиться, если расходится метод простых итераций, и наоборот.

Плюсом метода Зейделя, как и метода простых итераций, является его "самоисправляемость".

Метод Зейделя имеет преимущества перед методом простых итераций, так как он всегда сходится для нормальных систем линейных алгебраических уравнений, т.е. таких систем, в которых матрица A является симметрической и положительно определенной.

3.5 Метод верхней релаксации

Среди явных одношаговых итерационных методов наибольшее распространение получил метод верхней релаксации. Это связано с тем, что метод верхней релаксации содержит свободный параметр ω , изменяя который можно получать различную скорость сходимости итерационного процесса.

Форма записи метода верхней релаксации является обобщенной формулой метода Зейделя, в котором параметр $\omega = 1$:

$$x_i^{(k+1)} = (1 - \omega)x_i^{(k)} - \omega \left(\sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} + \sum_{j=i}^n a_{ij}x_j^{(k)} + \beta_i \right)$$

При этом для выполнения метода именно верхней релаксации, требуется выбрать параметр $\omega > 1$. Кроме того, необходимым условием схождения метода является $\omega \in (0, 2)$. Если матрица коэффициентов A симметрична и положительно определена, то это условие является достаточным.

В общем случае нет аналитической формулы для вычисления оптимального параметра $\omega_{\text{опт}}$, обеспечивающего наилучшую сходимость. Его значение может быть

получено либо экспериментальным путем, либо аналитически, но только для некоторых задач.

Сам алгоритм метода релаксации аналогичен тому, что описан для метода Зейделя, за исключением рекуррентной формулы, которая используется для уточнения очередного решения.

3.6 Метод сопряженных градиентов

Метод сопряженных градиентов – численный метод решения систем линейных алгебраических уравнений.

Пусть дана система n линейных уравнений $Ax = b$, где A симметричная положительно определённая матрица размером $n \times n$, а x и b – вектора размера n .

Тогда процесс решения СЛАУ можно реализовать как минимизацию следующего функционала: $F(x) = (Ax, x) + 2(b, x) \rightarrow \min$.

В самом деле, функция $F(x)$ достигает своего минимального значения тогда и только тогда, когда ее градиент $\nabla F(x) = Ax - b$ обращается в ноль. Таким образом, решение системы линейных уравнений можно искать как решение задачи безусловной минимизации.

Рассмотрим сам алгоритм метода сопряженных градиентов.

Итерационный процесс начинается с указания некоторого произвольного приближения x , а также нахождения вектора невязки r и вектора направления h (в дальнейшем – градиент). В результате подготовительного шага получаем:

$$x = 1 - \text{единичный вектор}, \quad r = h = b - Ax$$

Основные шаги выполняются последовательно и представляют собой следующий набор действий:

- В процессе очередной итерации вычисляется значение скалярного шага $alpha$:

$$alpha = \frac{(r, r)}{(Ah, h)}$$

- Находим очередное приближение:

$$x = x - alpha * h$$

- Обновляем значение вектора невязки, которое будет использоваться на следующей итерации:

$$r_{next} = r - alpha * Ah$$

- Вычисляем коэффициент β , соответствующий выполнению условия сопряженности $(Ah_{next}, h) = 0$, где h_{next} – значение градиента, которое будет получено для выполнения следующего шага:

$$\beta = \frac{(r_{next}, r_{next})}{(r, r)}$$

- Находим новое значение градиента:

$$h = r_{next} + \beta * h$$

Из анализа расчетных формул метода видно, что на каждой итерации требуется выполнить умножение матрицы A на вектор h , несколько скалярных произведений векторов и несколько операций над векторами.

Известно, что умножение матрицы на вектор имеет квадратичную сложность $O(n^2)$, а сложность умножения векторов или выполнения операций с ними – $O(n)$. Следовательно, сложность выполнения одной итерации можем оценить сверху, как $O(n^2)$.

Кроме того, известно, что для нахождения точного решения системы линейных уравнений с положительно определенной симметричной матрицей необходимо выполнить не более n итераций, тем самым, сложность всего метода составляет $O(n^3)$.

3.7 Метод LU-разложения

Метод LU-разложения опирается на возможность представления квадратной матрицы A в виде произведения двух треугольных матриц, а именно $A = LU$, где L – нижняя, а U – верхняя треугольные матрицы.

$$L = \begin{pmatrix} l_{11} & 0 & \dots & 0 \\ l_{21} & l_{22} & \dots & 0 \\ \dots & \dots & \dots & \dots \\ l_{n1} & l_{n2} & \dots & l_{nn} \end{pmatrix}, \quad U = \begin{pmatrix} 1 & u_{12} & \dots & u_{1n} \\ 0 & 1 & \dots & u_{2n} \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 1 \end{pmatrix}$$

Из общего вида элемента произведения $A = LU$, а также структуры матриц L и U следуют формулы для определения элементов этих матриц:

$$l_{ij} = a_{ij} - \sum_{s=1}^{j-1} l_{is}u_{sj}, \quad i \geq j, \quad u_{ij} = \frac{1}{l_{ii}} \left(a_{ij} - \sum_{s=1}^{i-1} l_{is}u_{sj} \right), \quad i < j$$

Используя разложение $A = LU$, систему $Ax = b$ можно представить в виде $LUx = b$. Решение такой системы сводится к последовательному решению двух простых систем с треугольными матрицами.

Алгоритм метода представляет собой следующий набор действий:

1. Выполнить разложение матрицы A , получить матрицы L и U .

2. Прямой ход. Произведение Ux обозначим через y . В результате решения системы $Ly = b$ находится вектор y .
3. Обратный ход. В результате решения системы $Ux = y$ находится решение задачи – столбец x .

В силу треугольности матриц L и U решения обеих систем находятся рекуррентно (как в обратном ходе метода Гаусса).

4. Сравнение результатов

Вычислительные эксперименты для сравнения описанных в данной работе методов решения систем линейных уравнений проводились на оборудовании со следующей аппаратной конфигурацией:

- Процессор: Intel Core i5-7200U, 2700 MHz,
- Оперативная память: 6012 МБ (DDR4 SDRAM), 2400 MHz,
- ОС: Microsoft Windows 10 Home, версия 10.0.18363 сборка 18383.

В качестве исходных данных для проведения экспериментов была выбрана система размерности 12 с симметричной положительно определённой матрицей коэффициентов, для которой выполняется свойство диагонального преобладания.

Таблица 1 Сравнение методов решения СЛАУ

Метод решения	Время, микросекунды	Число итераций
Метод Гаусса	15	Не является итерационным
Метод Крамера	120	Не является итерационным
Метод простых итераций	12	7
Метод Зейделя	9	3
Метод верхней релаксации	17	3
Метод LU-разложения	10	Не является итерационным
Метод сопряженных градиентов	19	9

По результатам, представленным в таблице можно сделать конкретные выводы лишь о прямых методах решения. Так, метод Гаусса значительно выигрывает у метода Крамера, т.к. в последнем случае большое число ресурсов тратится на вычисление определителей. Метод LU-разложения во многом схож с методом Гаусса, поэтому и время его работы сравнимо с результатами, полученными по Гауссу.

Насчет итерационных методов можно сказать лишь то, что, например, метод Зейделя, который в некотором смысле является улучшением метода простых итераций, оказался в данном конкретном случае действительно эффективнее. А метод простых итераций, в свою очередь, дал решение системы быстрее, чем прямой метод Гаусса.

Однако итерационные методы во многом зависят от выбора начального приближения, поэтому дать объективную сравнительную характеристику нельзя. Каждый из них хорош лишь при определенном наборе начальных условий.

5. Пример работы программы

При запуске программы перед пользователем появляется интерфейс управления.

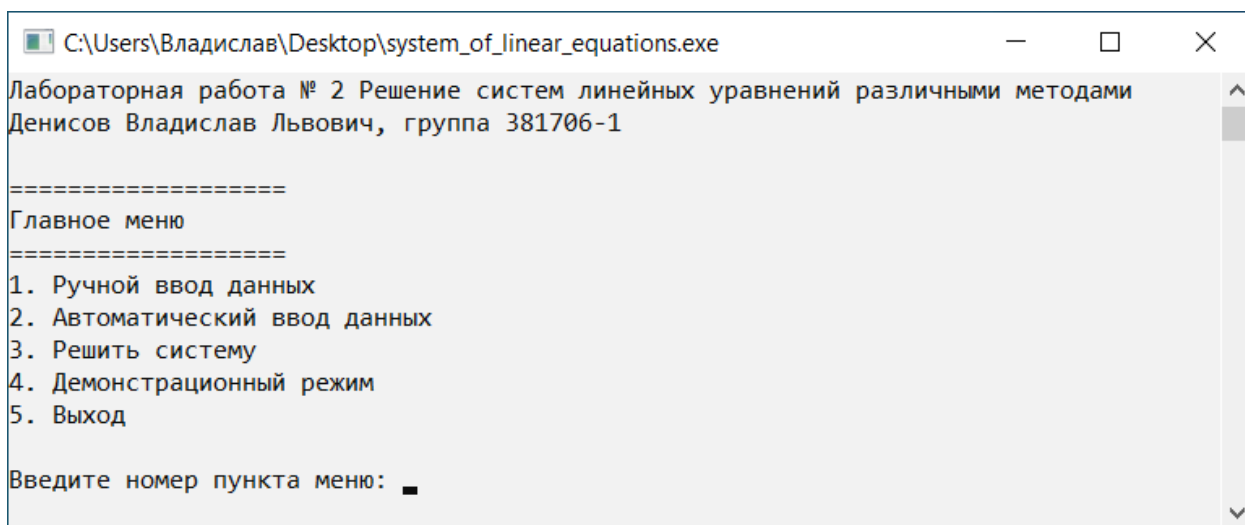


Рисунок 4 Главное меню программы

Рассмотрим пример, с автоматической генерацией системы линейных уравнений.

Вводим номер соответствующего пункта.

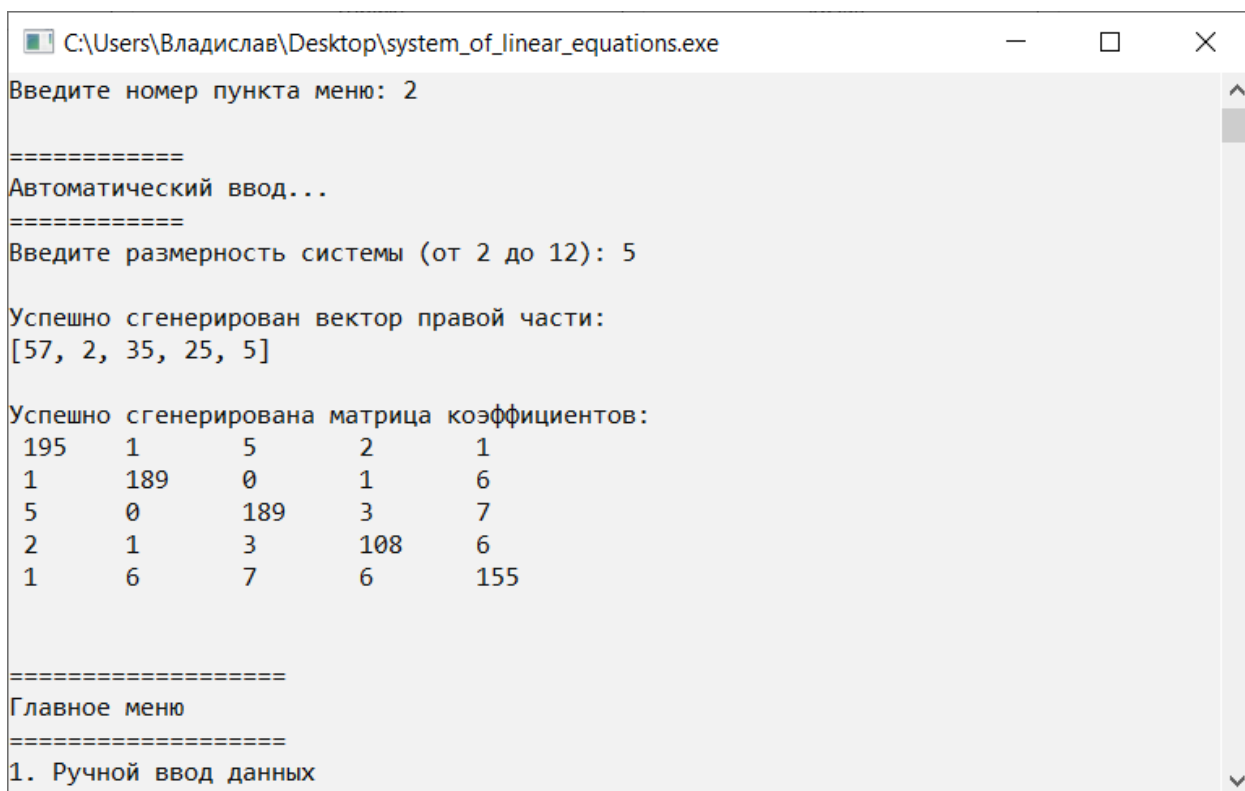


Рисунок 5 Режим автоматического ввода входных данных

На экране появились результаты автоматической генерации, а также возвращено главное меню. Затем перейдем в раздел решения системы. Вводим номер пункта 3. Перед нами откроется меню выбора способа решения.

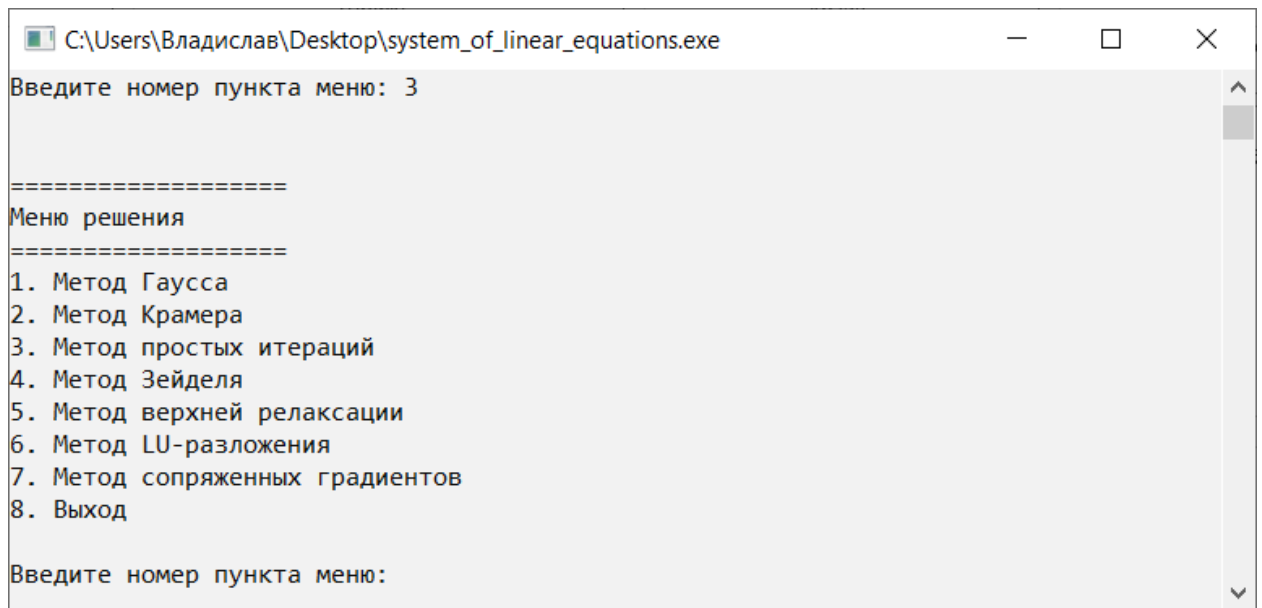


Рисунок 6 Меню выбора способа решения

Выберем решение системы при помощи метода Зейделя. Введём требуемую точность и получим результаты на консоль.

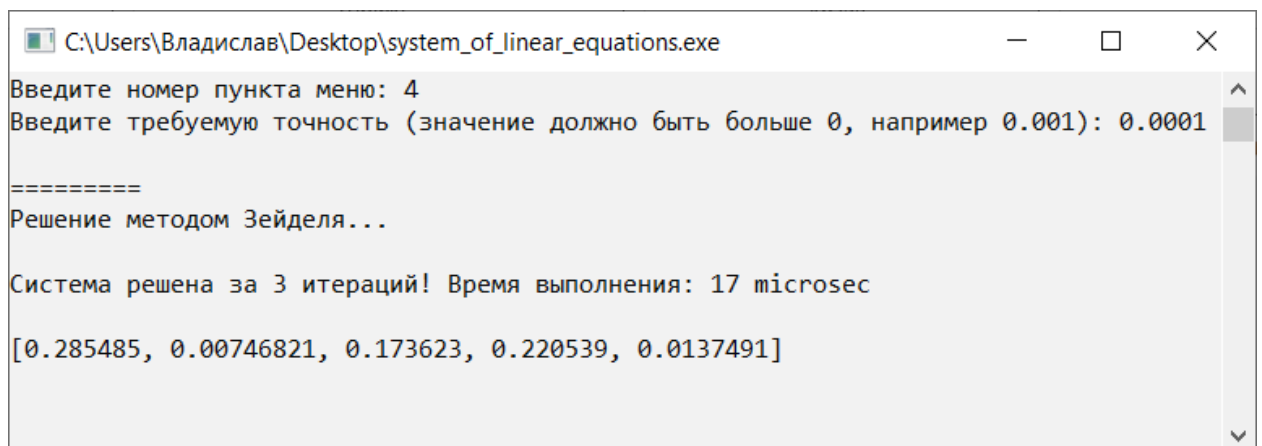


Рисунок 7 Решение системы линейных уравнений методом Зейделя

6. Заключение

В результате лабораторной работы разработан программный комплекс, который позволяет решать систему линейных уравнений, путем использования различных методов.

Разработанная программа позволяет задать матрицу коэффициентов системы линейных уравнений, а также вектор свободных членов вручную или же сгенерировать автоматически. Есть возможность выбора способа решения из списка всех тех, что были описаны в лабораторной работе.

Подробное рассмотрение различных методов решения систем линейных уравнений позволило лучше разобраться в этой теме. А проведение сравнения друг с другом времени их работы дало информацию о том, что итерационные методы дают хорошие результаты только при удачном выборе начальных условий.

Цели, поставленные в лабораторной работе, успешно достигнуты.

7. Литература

1. Костомаров, Д.П. Вводные лекции по численным методам: Учеб. пособие / А.П. Фаворский, Д.П. Костомаров .— М. : Логос, 2006 .— 130 с.
2. Самарский А.А. Введение численные методы. – СПб.: Лань, 2005.
3. Баркалов К.А. Методы параллельных вычислений. Н. Новгород: Изд-во Нижегородского госуниверситета им. Н.И. Лобачевского, 2011