Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное   
учреждение высшего образования

Национальный исследовательский Нижегородский государственный университет им. Н.И. Лобачевского

Институт информационных технологий, математики и механики

**Отчет по лабораторной работе**

**«Метод Гаусса»**

**Выполнил**:

студент группы 3821Б1-ПМ2

Сятов Никита Андреевич

**Проверил**:

преподаватель каф. МОСТ,

Волокитин В.Д.

Нижний Новгород

2021

**Содержание**

[Постановка задачи 3](#_Toc26962562)

[Метод решения 4](#_Toc26962563)

[Руководство пользователя 6](#_Toc26962564)

[Описание программной реализации 7](#_Toc26962565)

[Подтверждение корректности 8](#_Toc26962566)

[Заключение 9](#_Toc26962568)

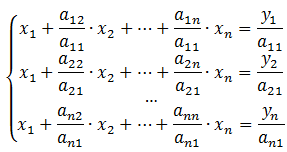
[Приложение 10](#_Toc26962569)

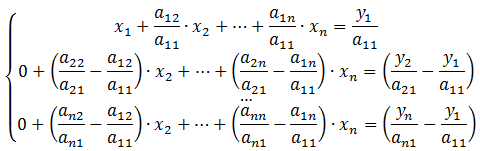
# Постановка задачи

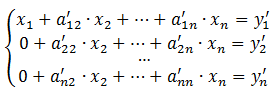
В данной лабораторной работе была цель реализовать метод Гаусса, а также реализовать класс вектор, класс квадратной матрицы N\*N, отнаследованной от вектора векторов при использовании шаблонов. Также реализовать класс СЛАУ с методом solve, который решает систему линейных уравнений методом Гаусса с ведущим элементом. Сделать проверку, использовать исключения для выявления деления на ноль, выхода за границы массивов и т. д.

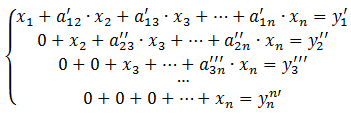
# Метод решения

Я использовал метод Гаусса с ведущим элементом. Его смысл заключается в том, чтобы преобразовать расширенную матрицу в верхний треугольный (ступенчатый) вид. Для этого используется последовательное исключение: последовательно исключаются по одной переменной, пока не останется одно уравнение с одной переменной в левой части. Для этого выбирают ненулевой максимальный элемент и делят строку на него, остальные строки делят на элемент, стоящий в том же столбце. После вычитаем из строки выбранную строку с максимальным элементом. Таким образом, мы приходим к нужной форме матрицы и делаем обратную подстановку.

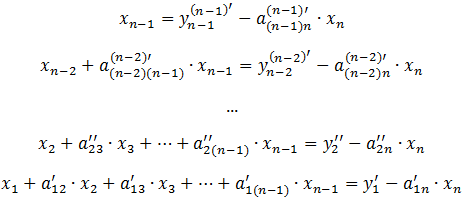


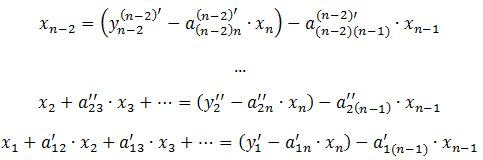






Обратная подстановка:





# Руководство пользователя

Пользователь должен ввести количество переменных и указать способ заполнения матрицы: рандомно или вручную пользователем. После создастся объект класса SLAU размером N\*N и заполнится выбранным способом. В консоль выведется результат

(х1, … , хN), если погрешность окажется меньше выбранного EPSILON, то в консоль выведется, что получено верное решение, если же погрешность окажется больше, то выведется погрешность и сообщение, что получено неверное решение.

# Описание программной реализации

Проект содержит классы: myVector<T>, myMatrix<T>(наследуется от myVector<myVector<T>>), SLAU<T>(наследуется от myMatrix<T>). Определён макрос EPSILON – максимальная погрешность. В myVector перегружены операции [] =, реализован метод resize для изменения размера вектора. В myMatrix реализован метод перемножения матрицы на вектор для проверки результата. В SLAU перегружен оператор =, реализованы методы: solve(решение системы методом Гаусса), revers\_motion(обратный ход и вывод результирующего вектора), check(подсчёт погрешности, проверка корректности).

# Подтверждение корректности

Для подтверждения корректности в программе используется метод check (класс SLAU). В нём вызывается матричный метод перемножения исходной матрицы на результирующий вектор, после чего вычисляется погрешность при помощи вычитания результирующего из исходного вектора значений b и сравнение с EPSILON, если разность по модулю превышает фиксируемое значение, то в консоль выводится погрешность и сообщение, что решение неверно.

# Заключение

Я реализовал метод Гаусса, использовав ООП. Создал класс матрица, отнаследованный от вектора векторов, подтвердил корректность программы. Увидел, что таким образом можно решать достаточно большие матрицы.

# Приложение

