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 для квадратной матрицы»**

**Выполнил**:

студент группы 382003-1

Линев Евгений Алексеевич

**Проверил**:

ассистент каф. МОСТ,

Волокитин В.Д.

Нижний Новгород

2019

**Содержание**

[Постановка задачи 3](#_Toc26962562)

[Метод решения 4](#_Toc26962563)

[Руководство пользователя 5](#_Toc26962564)

[Описание программной реализации 6](#_Toc26962565)

[Подтверждение корректности 7](#_Toc26962566)

[Результаты экспериментов 8](#_Toc26962567)

[Заключение 9](#_Toc26962568)

[Приложение 10](#_Toc26962569)

# Постановка задачи

Требуется реализовать метод Гаусса для действительной квадратной матрицы с выбором ведущего элемента. Для этого нужно:

* Реализовать шаблонный класс вектор;
* Реализовать класс матрица, которая является шаблоном класса вектор от вектора;
* Реализовать класс ( или метод класса матрица) Gausse, являющийся методом Гаусса. Метод Гаусса на вход принимает правую часть в качестве аргумента;
* Вывести результат в виде вектора.

# Метод решения

Для того, чтобы решить систему линейных уравнений, в данном случае требуется привести матрицу к диагональному виду. Для этого понадобится найти в первом столбце максимальный элемент и поставить строку, содержащую этот элемент, на первое место. Затем нужно разделить всю строку на значение этого элемента, так у нас в последствии по диагонали будут стоять 1. После этого идем по всем строкам и с помощью арифметических операций над ними и первой строкой приводим все элементы первого столбца но не первой строки к 0.

Затем двинемся ко второму столбцу и будем рассматривать вторую строку. То есть второй диагональный элемент. Среди всех строк не выше второй будем искать максимум в этом столбце и снова сделаем так, чтобы диагональный элемент был максимальным элементом в столбце. Снова приведем все элементы этого столбца кроме диагонального к нулю.

Сделаем так со всеми столбцами. С правой частью, разумеется, ведутся соответствующие преобразования.

Если у нас получится диагональная матрица, то можно вывести решение.

Если вдруг среди диагональных элементов будет 0, то есть два варианта событий:

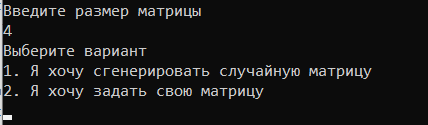
1. Если существует нулевая строка но соответствующий ей элемент правой части не равен нулю, то нет решений;
2. Иначе существует нулевая строка и соответствующий ей элемент правой части тоже равен нулю, тогда система линейно зависимая и решений бесконечно много.

# Руководство пользователя

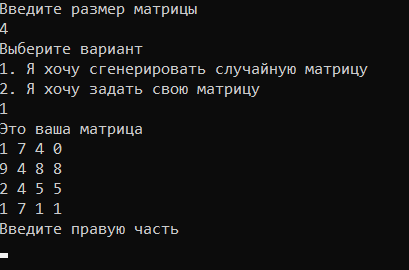
При запуске программы пользователю предлагается ввести размер квадратной матрицы, который также станет размером вектора b.



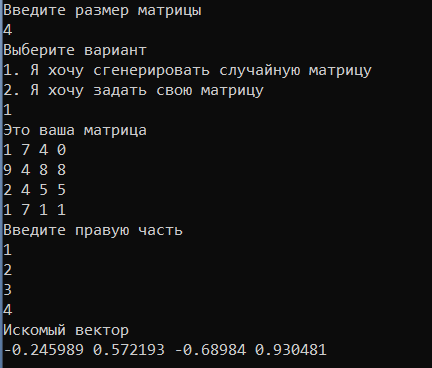
Затем предлагается ввести матрицу или сгенерировать случайную. Во втором случае она будет заполнена значениями типа int.



После этого вы должны задать правую часть. Ее случайное заполнение не предусмотрено.



После ввода правой части вы в скором времени получите результат.



# Описание программной реализации

В моей реализации метод Гаусса является методом класса Матрица и принимает вектор b в качестве аргумента. Создаем цикл с индексом j, который будет отвечать за столбцы матрицы. Это будет больший цикл, после каждой итерации которого мы будем получать столбец с нулями во всех строках в столбце j и единицу в j строке j столбце. Элемент j строки j столбца будем называть ведущим.

Затем заводим счетчик i, отвечающий за строки. Если элемент I строки j столбца будет больше элемента j строки j столбца более чем на величину, определяющую погрешность, то мы поменяем эти строки местами, а вместе с ними и значения вектора b, соответствующие этим строкам. Таким образом в j строке окажется наибольший элемент среди всех элементов j столбца. Цикл со счетчиком I заканчивается.

Если этот элемент окажется нулем, то дальнейшие преобразования бессмысленны и мы переходим к следующему значению j, то есть к следующему диагональному элементу.

Если нет, то делим всю j-ую строку ведущий элемент, так чтобы он стал единицей. Соответствующий элемент вектора b также делим на это число.

Создаем цикл со счетчиком i, отвечающим за строку. Если мы имеем дело не с диагональным элементом или элемент I строки j столбца равен нулю, то переходим к следующей итерации по i. Иначе заводим счетчик k=j, который будет отвечать за столбцы. С его помощью можно будет проходить по всем элементам строки и изменять их.

Умножаем всю строку j на значения элемента в I строке j столбца и затем из строки I вычитаем строку j. Сделать это поможет заведенный ранее счетчик k, с помощью которого каждый из элементов j строки сначала умножается на необходимое число, затем этот элемент вычитается из элемента того же столбца но I строки, а затем возвращает свое прежнее значение. Для вектора b делаются те же операции по завершению цикла со счетчиком k (так как там лишь один элемент в строке этот счетчик нам уже не нужен).

Завершается цикл со счетчиком I.

Завершается цикл со счетчиком j.

В итоге этих преобразований получилась некая матрица, которая теперь должна быть проверена на некоторые условия.

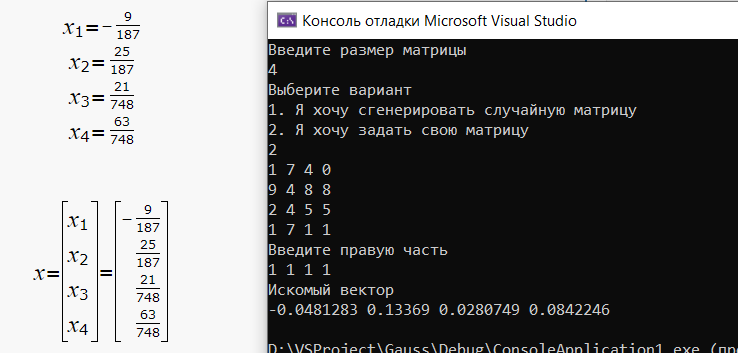
Проверяем диагональные элементы. Если какой-то среди них равен нулю, то проверяем всю строчку на количество нулей. Если вся строчка нулевая и соответствующий элемент вектора b не ноль, то система не имеет решений и программа завершается. Если находится нулевая строка и соответствующий элемент вектора b ноль, то заранее объявленная переменная (флаг) становится равна 1 и, если не будет найдена ни одна «нерешаемая» строка, то с помощью этого флага можно будет вывести, что решений бесконечно много.

Если нулей не обнаружено, то матрица диагональная и, соответственно, выводится требуемый вектор.

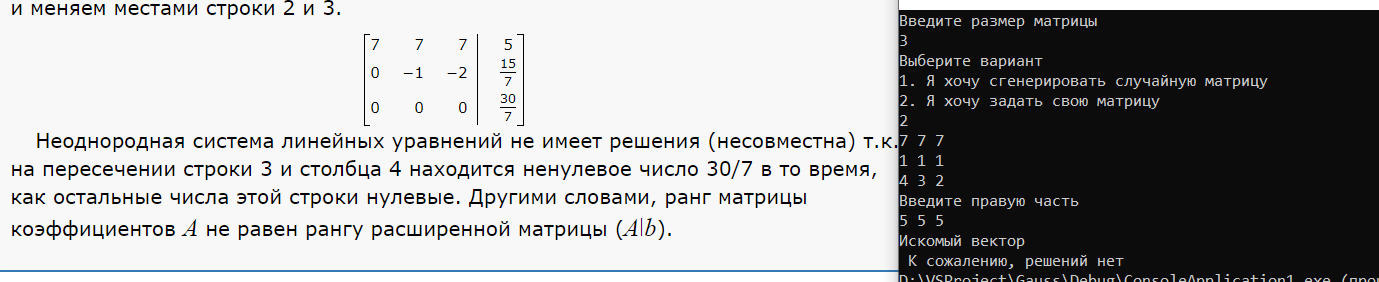
# Подтверждение корректности

Для подтверждения корректности выходных данных я использовал несколько онлайн-ресурсов, которые также нужны для реализации метода Гаусса. Их было несколько, так как меня интересовала точность полученных вычислений. При проверке на разных сайтах я получал то же решение, что и в своей программе, с учетом погрешности вычислений. Скрины экспериментов предоставлены в разделе «Результаты эксперементов».

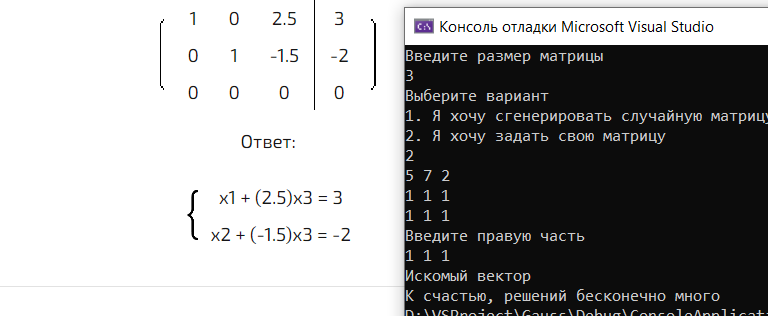
# Результаты экспериментов



Пример 1. Искомый вектор получен.



Пример 2. Решений нет.



Пример 3. Решений бесконечно много.

По данным экспериментов видно, что все три варианта учтены и совпадают с результатами, полученными с помощью онлайн-ресурсов.

# Заключение

Был реализован шаблонный класс вектор. Был реализован шаблонный класс матрица, являющийся шаблонам класса вектор от вектора. В классе матрица был реализован метод Gausse, который выполняет метод Гаусса, принимает на вход правую часть и возвращает ответ в виде вектор.

Для каждого класса были созданы три конструктора, а также диструктор, а также переопределены некоторые операторы. В частности – ввода и вывода.

Был создан хороший и понятный интерфейс, рассчитанный на пользователя.

Проверка показала, что программа работает корректно, с учетом погрешности (10^(-6)).

# Приложение

