Міністерство освіти та Науки України

Національний технічний університет “Харківський політехнічний інститут”

Кафедра «Програмна інженерія і інформаційні технології управління»

Індивідуальне завдання №1

з дисципліни

«Чисельні методи»

Виконали:

студенти групи КН-34б

Рузняєв Д.А.

Костюк I.Ю.

Перевірив:

Проф. Гужва В.О.

Харків

2016

**МЕТОДЫ РЕШЕНИЯ СЛАУ**

**Метод Гаусса** — классический метод решения [системы линейных алгебраических уравнений](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D0%BB%D0%B8%D0%BD%D0%B5%D0%B9%D0%BD%D1%8B%D1%85_%D0%B0%D0%BB%D0%B3%D0%B5%D0%B1%D1%80%D0%B0%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D1%85_%D1%83%D1%80%D0%B0%D0%B2%D0%BD%D0%B5%D0%BD%D0%B8%D0%B9) (СЛАУ). Это метод последовательного исключения [переменных](https://ru.wikipedia.org/wiki/%D0%9F%D0%B5%D1%80%D0%B5%D0%BC%D0%B5%D0%BD%D0%BD%D0%B0%D1%8F_%D0%B2%D0%B5%D0%BB%D0%B8%D1%87%D0%B8%D0%BD%D0%B0), когда с помощью элементарных преобразований система уравнений приводится к равносильной системе треугольного вида, из которой последовательно, начиная с последних (по номеру), находятся все переменные системы.

\left\{\begin{array}{lcr}
a_{11}x_1+\ldots+a_{1n}x_n &=& b_1 \\
\ldots & & \\
a_{m1}x_1+\ldots+a_{mn}x_n &=& b_m \\
\end{array}\right.\iff Ax = b,\quad A=\left( \begin{array}{ccc}
a_{11} & \ldots & a_{1n}\\
\ldots &  &  \\
a_{m1} & \ldots & a_{mn}
\end{array}\right),\quad x = \left( \begin{array}{c}x_1 \\ \vdots \\ x_n \end{array} \right), \quad b = \left( \begin{array}{c}b_1 \\ \vdots \\ b_m \end{array} \right).\quad (1)Пусть исходная система выглядит следующим образом.

Матрица A называется основной матрицей системы, b — столбцом свободных членов.

Тогда, согласно свойству [элементарных преобразований](https://ru.wikipedia.org/wiki/%D0%AD%D0%BB%D0%B5%D0%BC%D0%B5%D0%BD%D1%82%D0%B0%D1%80%D0%BD%D1%8B%D0%B5_%D0%BF%D1%80%D0%B5%D0%BE%D0%B1%D1%80%D0%B0%D0%B7%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F_%D0%BC%D0%B0%D1%82%D1%80%D0%B8%D1%86%D1%8B) над строками, основную матрицу этой системы можно привести к ступенчатому виду (эти же преобразования нужно применять к столбцу свободных членов):

\left\{\begin{array}{rcl}
\alpha_{1j_1}x_{j_1}+\alpha_{1j_2}x_{j_2}+\ldots+\alpha_{1j_r}x_{j_r}+\ldots+\alpha_{1j_n}x_{j_n} &=& \beta_1 \\
                     \alpha_{2j_2}x_{j_2}+\ldots+\alpha_{2j_r}x_{j_r}+\ldots+\alpha_{2j_n}x_{j_n} &=& \beta_2 \\
                                                                                               &\ldots& \\
                                                 \alpha_{rj_r}x_{j_r}+\ldots+\alpha_{rj_n}x_{j_n} &=& \beta_r \\
                                                                                                0 &=& \beta_{r+1} \\
                                                                                               &\ldots& \\
                                                                                                0 &=& \beta_m
\end{array}\right.,\qquad \alpha_{1j_1},\ldots,\alpha_{rj_r}\neq 0.

При этом будем считать, что [базисный минор](https://ru.wikipedia.org/wiki/%D0%9C%D0%B8%D0%BD%D0%BE%D1%80_%28%D0%BB%D0%B8%D0%BD%D0%B5%D0%B9%D0%BD%D0%B0%D1%8F_%D0%B0%D0%BB%D0%B3%D0%B5%D0%B1%D1%80%D0%B0%29) (ненулевой [минор](https://ru.wikipedia.org/wiki/%D0%9C%D0%B8%D0%BD%D0%BE%D1%80_%28%D0%BB%D0%B8%D0%BD%D0%B5%D0%B9%D0%BD%D0%B0%D1%8F_%D0%B0%D0%BB%D0%B3%D0%B5%D0%B1%D1%80%D0%B0%29) максимального порядка) основной матрицы находится в верхнем левом углу, то есть в него входят только коэффициенты при переменных x_{j_1},\ldots,x_{j_r}\!.

Тогда переменные x_{j_1},\ldots,x_{j_r}\!называются *главными переменными*. Все остальные называются *свободными*.

Если хотя бы одно число \beta_i\neq 0\!, где i > r, то рассматриваемая система [несовместна](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D0%BB%D0%B8%D0%BD%D0%B5%D0%B9%D0%BD%D1%8B%D1%85_%D0%B0%D0%BB%D0%B3%D0%B5%D0%B1%D1%80%D0%B0%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D1%85_%D1%83%D1%80%D0%B0%D0%B2%D0%BD%D0%B5%D0%BD%D0%B8%D0%B9), т.е. у неё нет ни одного решения.

Пусть \beta_i = 0\!для любых i > r.

\left\{\begin{array}{rcc}
x_{j_1}+\widehat{\alpha}_{1j_2}x_{j_2}+\ldots+\widehat{\alpha}_{1j_r}x_{j_r}&=& \widehat{\beta}_1-\widehat{\alpha}_{1j_{r+1}}x_{j_{r+1}}-\ldots- \widehat{\alpha}_{1j_n}x_{j_n} \\
                     x_{j_2}+\ldots+\widehat{\alpha}_{2j_r}x_{j_r}&=& \widehat{\beta}_2-\widehat{\alpha}_{2j_{r+1}}x_{j_{r+1}}-\ldots- \widehat{\alpha}_{2j_n}x_{j_n} \\
                                                           &\ldots& \\
                                                       x_{j_r}&=& \widehat{\beta}_r-\widehat{\alpha}_{rj_{r+1}}x_{j_{r+1}}-\ldots- \widehat{\alpha}_{rj_n}x_{j_n} \\
\end{array}\right., \qquad \widehat{\beta}_i=\frac{\beta_i}{\alpha_{ij_i}},\quad \widehat{\alpha}_{ij_k}=\frac{\alpha_{ij_k}}{\alpha_{ij_i}}\quad (2)Перенесём свободные переменные за знаки равенств и поделим каждое из уравнений системы на свой коэффициент при самом левом x\!(\alpha_{ij_i},\, i=1,\ldots,r\!, где i\! — номер строки):

i=1,\ldots,r,\quad k=i+1,\ldots,n.\!где,

Если свободным переменным системы (2) придавать все возможные значения и решать новую систему относительно главных неизвестных снизу вверх (то есть от нижнего уравнения к верхнему), то мы получим все решения этой [СЛАУ](https://ru.wikipedia.org/wiki/%D0%A1%D0%9B%D0%90%D0%A3). Так как эта система получена путём [элементарных преобразований](https://ru.wikipedia.org/wiki/%D0%AD%D0%BB%D0%B5%D0%BC%D0%B5%D0%BD%D1%82%D0%B0%D1%80%D0%BD%D1%8B%D0%B5_%D0%BF%D1%80%D0%B5%D0%BE%D0%B1%D1%80%D0%B0%D0%B7%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F_%D0%BC%D0%B0%D1%82%D1%80%D0%B8%D1%86%D1%8B) над исходной системой (1), то по теореме об эквивалентности при элементарных преобразованиях системы (1) и (2) эквивалентны, то есть множества их решений совпадают.

**Следствия:**  
1. Если в совместной системе все переменные главные, то такая система является определённой.

2. Если количество переменных в системе превосходит число уравнений, то такая система является либо неопределённой, либо несовместной.

**Критерий совместности**

Упомянутое выше условие \beta_i= 0\!для всех i > r\!может быть сформулировано в качестве необходимого и достаточного условия совместности:

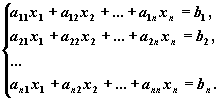
Напомним, что рангом совместной системы называется ранг её основной матрицы (либо расширенной, так как они равны).

Алгоритм решения [СЛАУ](https://ru.wikipedia.org/wiki/%D0%A1%D0%9B%D0%90%D0%A3) методом Гаусса подразделяется на два этапа.

* На первом этапе осуществляется так называемый прямой ход, когда путём [элементарных преобразований](https://ru.wikipedia.org/wiki/%D0%AD%D0%BB%D0%B5%D0%BC%D0%B5%D0%BD%D1%82%D0%B0%D1%80%D0%BD%D1%8B%D0%B5_%D0%BF%D1%80%D0%B5%D0%BE%D0%B1%D1%80%D0%B0%D0%B7%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F_%D0%BC%D0%B0%D1%82%D1%80%D0%B8%D1%86%D1%8B) над строками систему приводят к ступенчатой или [треугольной форме](https://ru.wikipedia.org/wiki/%D0%A2%D1%80%D0%B5%D1%83%D0%B3%D0%BE%D0%BB%D1%8C%D0%BD%D0%B0%D1%8F_%D0%BC%D0%B0%D1%82%D1%80%D0%B8%D1%86%D0%B0), либо устанавливают, что система несовместна. А именно, среди элементов первого столбца матрицы выбирают ненулевой, перемещают его на крайнее верхнее положение перестановкой строк и вычитают получившуюся после перестановки первую строку из остальных строк, домножив 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 были совершены, первую строку и первый столбец мысленно вычёркивают и продолжают пока не останется матрица нулевого размера. Если на какой-то из итераций среди элементов первого столбца не нашёлся ненулевой, то переходят к следующему столбцу и проделывают аналогичную операцию.
* На втором этапе осуществляется так называемый обратный ход, суть которого заключается в том, чтобы выразить все получившиеся базисные переменные через небазисные и построить [фундаментальную систему решений](https://ru.wikipedia.org/wiki/%D0%A4%D1%83%D0%BD%D0%B4%D0%B0%D0%BC%D0%B5%D0%BD%D1%82%D0%B0%D0%BB%D1%8C%D0%BD%D0%B0%D1%8F_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D1%80%D0%B5%D1%88%D0%B5%D0%BD%D0%B8%D0%B9), либо, если все переменные являются базисными, то выразить в численном виде единственное решение системы линейных уравнений. Эта процедура начинается с последнего уравнения, из которого выражают соответствующую базисную переменную (а она там всего одна) и подставляют в предыдущие уравнения, и так далее, поднимаясь по «ступенькам» наверх. Каждой строчке соответствует ровно одна базисная переменная, поэтому на каждом шаге, кроме последнего (самого верхнего), ситуация в точности повторяет случай последней строки.

АЛГОРИТМ ЧИСЛЕННОГО РЕШЕНИЯ СЛАУ МЕТОДОМ ГАУССА

Розглянемо систему лінійних алгебраїчних рівнянь (СЛАР) відносно n невідомих x1 , x2 , ..., xn:



Ця система в "згорнутому" вигляді може бути записана так:Sni=1aij xj = bi ,i=1,2, ..., n.

В соответствии с правилом умножения матриц рассмотренная система линейных уравнений может быть записана в матричной форме Ax=b.

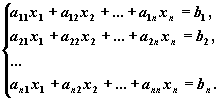
Матрица A, столбцами которой являются коэффициенты при соответствующих неизвестных, а строками - коэффициенты при неизвестных в соответствующем уравнении называется матрицей системы. Матрица-столбец b, элементами которой являются правые части уравнений системы, называется матрицей правой части или просто правой частью системы. Матрица-столбец **x**, элементы которой - искомые неизвестные, называется решением системы.

Система линейных алгебраических уравнений, записанная в виде Ax=b, является матричным уравнением.

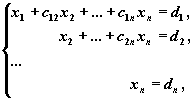
Если матрица системы невырождена, то у нее существует обратная матрица и тогда решение системы Ax=b дается формулой:

x=A-1 b.

Метод Гаусса применим для решения системы линейных алгебраических уравнений c невырожденной матрицей системы. Идея метода Гаусса состоит в том, что систему n линейных алгебраических уравнений относительно n неизвестных x1 , x2 , ..., xn



приводят последовательным исключением неизвестных к эквивалентной системе с треугольной матрицей

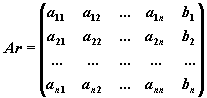


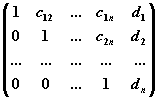
решение которой находят по рекуррентным формулам:

xn =dn , xi = di -S nk=i+1 cik xk , i=n-1, n-2, ...,1.

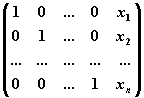
Матричная запись метода Гаусса.

1. **Прямой ход** метода Гаусса: приведение расширенной матрицы системы

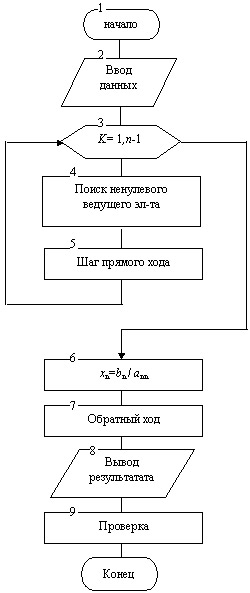
    
к ступенчатому виду

   
с помощью элементарных операций над строками матрицы:

* + перестановка строк;
  + умножение строки на число, отличное от нуля;
  + сложение строки матрицы с другой строкой, умноженной на отличное от нуля чиcло).

1. **Обратный ход** метода Гаусса: преобразование полученной ступенчатой матрицы к матрице, в первых n столбцах которой содержится единичная матрица ,   
   последний, (n+1)-й, столбец этой матрицы содержит решение системы. 

БЛОК-СХЕМА МЕТОДА ГАУССА



КОД ПРОГРАММЫ МЕТОДА ГАУССА

import java.util.Scanner;  
  
*/\*\*  
 \* Created by dmitr on 29.05.2016.  
 \*/*public class Gauss {  
 public static void main(String[] arg) {  
 int c, d;  
 @SuppressWarnings("resource")  
 Scanner s = new Scanner(System.*in*);  
 System.*out*.print("Количество переменных: ");  
 c = s.nextInt();  
 d = c + 1;  
 double[][] a = new double[c][d];  
  
 for (int i = 0; i < c; i++) {  
 System.*out*.print("Уравнение " + (i + 1) + "\n");  
 for (int j = 0; j < d; j++) {  
 if (j < d - 1) {  
 System.*out*.print("Коэффициент при Х[" + (j + 1) + "] ");  
 } else {  
 System.*out*.print("Равно: ");  
 }  
  
 a[i][j] = s.nextDouble();  
 }  
 }  
  
 System.*out*.print("Введенная система: \n");  
  
 for (int i = 0; i < c; i++) {  
 for (int j = 0; j < d; j++) {  
 if (j < d - 1) {  
 System.*out*.print(a[i][j] + "\*X" + (j + 1));  
 if (j < d - 2) {  
 System.*out*.print("+ ");  
 } else {  
 System.*out*.print(" ");  
 }  
 } else  
 System.*out*.print(" = " + a[i][j]);  
 }  
 System.*out*.println();  
 }  
  
 //прямой код  
 double x[] = new double[a.length];  
 double m;  
 for (int k = 1; k < a.length; k++) {  
 for (int j = k; j < a.length; j++) {  
 m = a[j][k - 1] / a[k - 1][k - 1];  
 for (int i = 0; i < a[j].length; i++) {  
 a[j][i] = a[j][i] - m \* a[k - 1][i];  
 }  
 x[j] = x[j] - m \* x[k - 1];  
 }  
 }  
 //вывод преобразованного массива  
 System.*out*.println(" ");  
 for (int i = 0; i < c; i++) {  
 for (int j = 0; j < d; j++) {  
 if (j < d - 1) {  
 System.*out*.print(a[i][j] + "\*X" + (j + 1));  
 if (j < d - 2) {  
 System.*out*.print("+ ");  
 } else {  
 System.*out*.print(" ");  
 }  
 } else  
 System.*out*.print(" = " + a[i][j]);  
 }  
 System.*out*.println();  
 }  
  
 //Обратный ход  
 for (int i = a.length - 1; i >= 0; i--) {  
 m = 0;  
 for (int j = i + 1; j < a.length; j++)  
 m += a[i][j] \* x[j];  
  
 x[i] = (a[i][a.length] - m) / a[i][i];  
 }  
 //Вывод ответа  
 System.*out*.println("Ответ: ");  
 for (int i = 0; i < x.length; i++) {  
 System.*out*.println("X[" + (i + 1) + "]" + " = " + x[i]);  
 }  
 }  
  
  
}

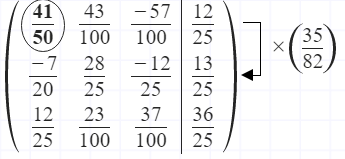
РУЧНОЙ ПРОСЧЕТ МЕТОДОМ ГАУССА

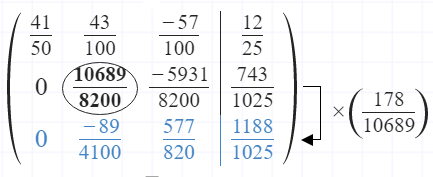
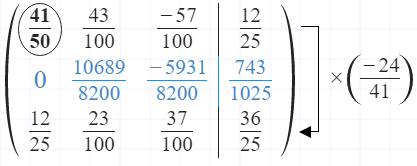
*Студента Рузняева Д.А.*

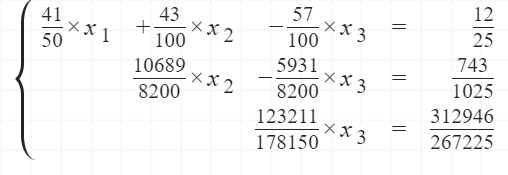
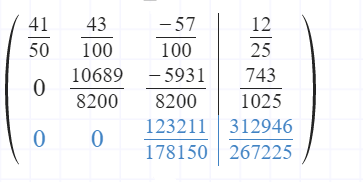
0,82\*х1+0,43\*х2-0,57\*х3 = 0,48

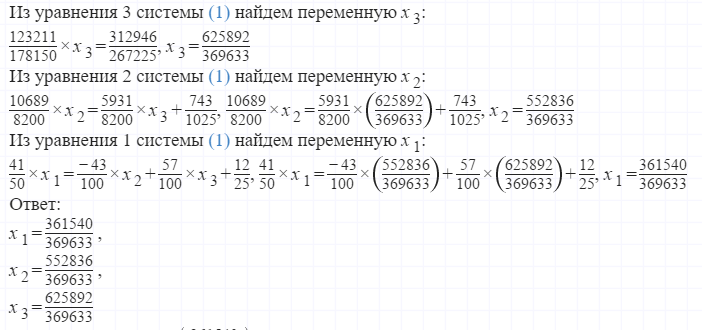
-0,35\*x1 + 1,12\*х-0.48\*x3 = 0.52

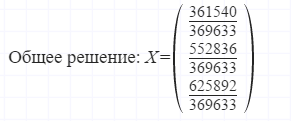
0.48\*x1+0.23\*x2+0.37x3=1.44











Х1 = 0,978; X2 = 1,495; X3 = 1,693

РЕШЕНИЕ СИСТЕМЫ С ПОМОШЬЮ ПРОГРАММЫ

МЕТОД ГАУССА

*Студента Рузняева Д.А.*

Введенная система:

0.82\*X1+ 0.43\*X2+ -0.57\*X3 = 0.48

-0.35\*X1+ 1.12\*X2+ -0.48\*X3 = 0.52

0.48\*X1+ 0.23\*X2+ 0.37\*X3 = 1.44

0.82\*X1+ 0.43\*X2+ -0.57\*X3 = 0.48

0.0\*X1+ 1.30353658\*X2+ -0.72329268\*X3 = 0.72487804

5.55111512E-17\*X1+ -3.46944695E-18\*X2+ 0.69161380\*X3 = 1.171095518

Ответ:

X[1] = 0.97810530

X[2] = 1.49563485

X[3] = 1.69327955

РУЧНОЙ ПРОСЧЕТ МЕТОДОМ ГАУССА

*Студента Костюка И.Ю.*

0,1\*x1 + 12\*x2 – 0,13\*x3 = 0,1

0,12\*x1 + 0,71\*x2 + 0,15\*x3 = 0,26

-0,13\*x1 + 0,15\*x2 + 0,3\*x3 = 0,38

**Приводим систему к необходимому виду:**

-0,13\*x1 + 0,15\*x2 + 0,3\*x3 = 0,38 (1)

0,12\*x1 + 0,71\*x2 + 0,15\*x3 = 0,26 (2)

0,1\*x1 + 12\*x2 – 0,13\*x3 = 0,1 (3)

Прямой ход:

**Разделим уравнение (1) на коэффициент при х1:**

x1 – 1,1538\*x2 – 4,8461\*x3 = -2,923

**Умножим полученное уравнение на коэффициент при уравнении (2) и (3) и после сложим со (2) и (3) уравнением соответственно, после чего получим систему следующего вида:**

-0,13\*x1 + 0,15\*x2 + 0,3\*x3 = 0,38

0,8484\*x2 + 0,7315\*x3 = 0,6107

12,11538\*x2 + 0,35461\*x3 = 0,3923

**Приведем систему к необходимому виду:**

-0,13\*x1 + 0,15\*x2 + 0,3\*x3 = 0,38 (1)

12,11538\*x2 + 0,35461\*x3 = 0,3923 (2)

0,8484\*x2 + 0,7315\*x3 = 0,6107 (3)

**Разделим уравнение (2) на коэффициент при х2:**

x2 + 0,02927\*x3 = 0,03238

**Умножим полученное уравнение на коэффициент при х2 из уравнения (3) и вычтем уравнение (3):**

0,7067\*x3 = 0,5832

**В результате прямого хода получили систему:**

-0,13\*x1 + 0,15\*x2 + 0,3\*x3 = 0,38 (1)

12,11538\*x2 + 0,35461\*x3 = 0,3923 (2)

0,7067\*x3 = 0,5832 (3)

Обратный ход:

**Из уравнения (3) найдем х3:**

х3 = 0,825284

**Подставив х3 в уравнение (2) найдем х2:**

х2 = 0,008224

**Подставив х2, х3 в уравнение (1) найдем х1:**

х1 = 1,0863

**Система решена, ответ:**

х1 = 1,0863

х2 = 0,008224

х3 = 0,825284

РЕШЕНИЕ СИСТЕМЫ С ПОМОЩЬЮ ПРОГРАММЫ

МЕТОД ГАУССА

*Студента Костюка И.Ю.*

Введенная система:

0.1\*X1+ 12.0\*X2+ -0.13\*X3 = 0.1

0.12\*X1+ 0.71\*X2+ 0.15\*X3 = 0.26

-0.13\*X1+ 0.15\*X2+ 0.3\*X3 = 0.38

0.1\*X1+ 12.0\*X2+ -0.13\*X3 = 0.1

0.0\*X1+ -13.68999999\*X2+ 0.306\*X3 = 0.14

0.0\*X1+ 0.0\*X2+ 0.48304528\*X3 = 0.671066471

Ответ:

X[1] = 1.08634234

X[2] = 0.00822439

X[3] = 0.82528459

# **Метод Гаусса — Зейделя**

A=\left(
\begin{array}{ccc}
a_{11} & \ldots & a_{1n} \\
\vdots & \ddots & \vdots \\
a_{n1} & \ldots & a_{nn} 
\end{array} \right),\quad \vec{b}=\left(
\begin{array}{c}
b_1 \\
\vdots \\
b_n 
\end{array} \right)**Метод Гаусса—Зейделя** является классическим итерационным методом решения [системы линейных уравнений](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D0%BB%D0%B8%D0%BD%D0%B5%D0%B9%D0%BD%D1%8B%D1%85_%D0%B0%D0%BB%D0%B3%D0%B5%D0%B1%D1%80%D0%B0%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D1%85_%D1%83%D1%80%D0%B0%D0%B2%D0%BD%D0%B5%D0%BD%D0%B8%D0%B9).

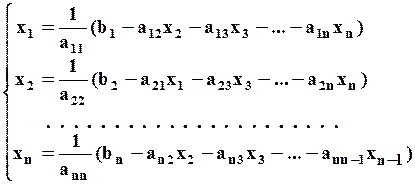
Возьмём систему: A\vec{x}=\vec{b}, где

\left\{
\begin{array}{rcl}
a_{11}x_1 + \ldots + a_{1n}x_n& = & b_{1} \\
& &\\
a_{n1}x_1 + \ldots + a_{nn}x_n & = & b_{n} 
\end{array} \right.

Или

И покажем, как её можно решить с использованием метода Гаусса-Зейделя.

Допустим, диагональные коэффициенты исходной матрицы {**aij**}отличны от нуля (в противном случае можно переставить местами уравнения исходной системы). Представим исходную систему в следующем виде:



Если теперь задать для неизвестных их начальные приближенные значения http://ok-t.ru/studopediaru/baza5/1780124706440.files/image153.gif**,** то система позволяет вычислить более точные значения неизвестных на первом шаге (на первой итерации):

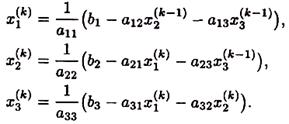


Используя найденные значения неизвестных**,** можно еще более уточнить их на второй итерации:



и так далее.

В данном методе для нахождения значения i-го неизвестного на каждой итерации используются значения предыдущих неизвестных, уже найденные на данной итерации. Общую формулу определения i-го неизвестного на k-й итерации для системы n уравнений можно записать так:



http://ok-t.ru/studopediaru/baza5/1780124706440.files/image159.gif…………………………………………………….

Итерационный процесс продолжается до тех пор, пока все значения **xi(k),**не станут достаточно близкими к **xi(k-1).** Близость этих значений можно охарактеризовать максимальной абсолютной величиной их разности **d**. Тогда при заданной точности вычислений **e > 0** критерий окончания итерационного процесса можно записать в виде

http://ok-t.ru/studopediaru/baza5/1780124706440.files/image163.gif

Доказано, что ***для сходимости итерационного процесса достаточно, чтобы модули диагональных коэффициентов для каждого уравнения были не меньше суммы модулей всех остальных коэффициентов:***

http://ok-t.ru/studopediaru/baza5/1780124706440.files/image165.gifhttp://ok-t.ru/studopediaru/baza5/1780124706440.files/image167.gif

При этом хотя бы для одного уравнения неравенство должно выполняться строго.

Итерационный процесс метода Гаусса-Зейделя отличается от метода простой итерации тем,что при решении системы вычисление (k+1) приближения xi, при i >1 используются уже ранее вычисленные (k+1) приближения неихвестных x1,…, xi-1 и k приближение неизвестных xi+1 ,…, xn

АЛГОРИТМ ЧИСЛЕННОГО РЕШЕНИЯ СЛАУ МЕТОДОМ ГАУССА-ЗЕЙДЕЛЯ

1.Проверить выполнение условия диагонального преобладания

http://ok-t.ru/studopediaru/baza5/1780124706440.files/image167.gifhttp://ok-t.ru/studopediaru/baza5/1780124706440.files/image165.gif

Если условие выполняется для любого i, то вычисляем элементы матрицы B и вектора d:

bij= - аij/aii  di= bi/aii при условии,что bii=0 i,j=1,2,…,n

2. Если условие диагонального преобладания не выполняется, то привести систему к виду удобному для итераций

3. Вычислить норму матрицы В

||B||1 = max ∑ |bij|

Если ||B||1 < 1, то итерационный процесс сходится (продолжается)

Если ||B||1 > 1, то итерационный процесс расходится (вычисления заканчиваются)

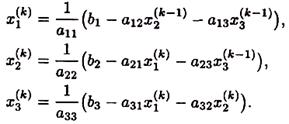
4. Вычислим норму вектора d

||d||1 = max |di|

5. Выбрать в качестве начального приближения вектор d, где х0 = d

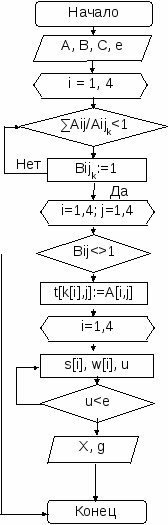
6. Вычислить необходимое количество итераций для достижения требуемой точности решения системы

7. Реализовать итерационный процесс в виде



и реализовать его k раз.

БЛОК-СХЕМА МЕТОДА ГАУССА - ЗЕЙДЕЛЯ



КОД ПРОГРАММЫ МЕТОДА ГАУССА-ЗЕЙДЕЛЯ

package gaussZeidel;  
  
import javafx.print.Printer;  
  
import java.io.PrintWriter;  
import java.util.Locale;  
import java.util.Scanner;  
  
*/\*\*  
 \* Created by dmitr on 03.06.2016.  
 \*/*public class gaussZeidel {  
 public static void main(String[] args) {  
 Scanner scanner = new Scanner(System.*in*);  
 scanner.useLocale(new Locale("Russian"));  
 PrintWriter printWriter = new PrintWriter(System.*out*);  
 System.*out*.println("Введите количество переменных:");  
 int size;  
 size = scanner.nextInt();  
 double[][] matrix = new double[size][size + 1];  
 System.*out*.println("Введите коеффициенты: ");  
 for (int i = 0; i < size; i++) {  
 for (int j = 0; j < size + 1; j++) {  
 matrix[i][j] = scanner.nextDouble();}}  
  
 System.*out*.println("Введите точность:");  
 double eps;  
 eps = scanner.nextDouble();  
  
 double[] previousVariableValues = new double[size];  
 for (int i = 0; i < size; i++) {  
 previousVariableValues[i] = 0.0;  
 }  
  
 while (true) {  
 double[] currentVariableValues = new double[size];  
 for (int i = 0; i < size; i++) {  
 currentVariableValues[i] = matrix[i][size];  
 for (int j = 0; j < size; j++) {  
 if (j < i) {  
 currentVariableValues[i] -= matrix[i][j] \* currentVariableValues[j];}}  
 if (j > i) {  
 currentVariableValues[i] -= matrix[i][j] \* previousVariableValues[j];}}  
 currentVariableValues[i] /= matrix[i][i];  
 }  
 double err = 0.0;  
 for (int i = 0; i < size; i++) {  
 err += Math.*abs*(currentVariableValues[i] - previousVariableValues[i]);  
 }  
 if (err < eps) { break;  
 }  
 previousVariableValues = currentVariableValues;  
 }  
 printWriter.println("Ответ: ");  
 for (int i = 0; i < size; i++) {  
 printWriter.print("X[" + (i + 1) + "] = " + previousVariableValues[i] + "\n");}  
 scanner.close();  
 printWriter.close();}}

РУЧНОЙ ПРОСЧЕТ МЕТОДОМ ГАУССА – ЗЕЙДЕЛЯ

*Студента Рузняева Д.А.*

Исходные данные: ε = 0,001.

* 1. Проверяем есть ли диагональное преобладание:

5,4 < |-6,2| + |-0,5|

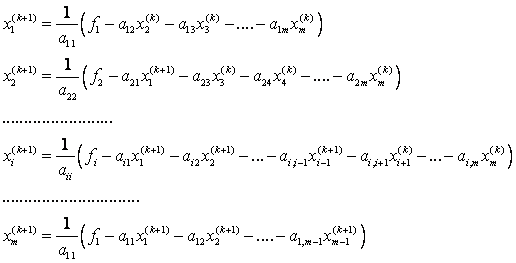
2,3 < |3,4|+ |0,8|

3,8 < |2,4| + |-1,1|

Ни в одном из уравнений нет диагонального преобладания. Значит необходимо выполнить преобразования системы.

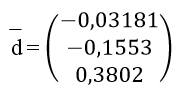
Сложим первое и второе уравнение, запишем ответ в первое; отнимем от первого второе, запишем ответ во второе уравнение; от первого отнимем третье, умноженное на 2, ответ запишем в третье уравнение.

Используя следующие преобразования



получим систему имеющую следующий вид:

Теперь вычислим элементы матрицы B и вектора d:



3. Вычислим норму матрицы B:

||В||1 = max ∑|bij|, j = 1,n.

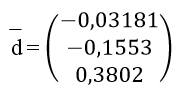
||В||1 = max {0,4773; 0,3882; 0,5679} = 0,5679 < 1 => процесс продолжается.

4. Вычислим норму вектора D:

||d||1 = max |di|, i=1,n.

||d||1 =0,3802.

5. Выберем в качестве начального приближения вектор d:



6. Вычислим необходимое число итераций для достижения требуемой точности решения системы:

K<= - 1

K<= – 1

K = 15

7. Реализовываем итерационный процесс:

1 итерация:

Новое приближение:

x1 =

2 итерация:

Новое приближение:

x2 =

3 итерация:

Ответ:

x3 =

РЕШЕНИЕ СИСТЕМЫ С ПОМОЩЬЮ ПРОГРАММЫ

МЕТОД ГАУССА-ЗЕЙДЕЛЯ

*Студента Рузняева Д.А.*

Введите точность: 0.001

Введенная система:

5.4\*X1- 6.2\*X2+ -0.5\*X3 = 0.52

3.4\*X1+ 2.3\*X2+ 0.8\*X3 = -0.8

2.4\*X1-1.1\*X2+ 3.8\*X3 = 1.8

0.82\*X1+ 0.43\*X2+ -0.57\*X3

Ответ:

X[1] = -0.16698105

X[2] = -0.27093485

X[3] = 0.50142955

РУЧНОЙ ПРОСЧЕТ МЕТОДОМ ГАУССА – ЗЕЙДЕЛЯ

*Студента Костюка И.Ю.*

Исходные данные: ε = 0,001.

1. Проверяем есть ли диагональное преобладание:

3,3 < |2,1 + 2,8|

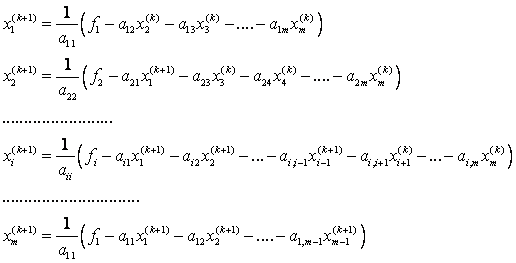
3,7 < |4,1 + 4,8|

1,1 < |2,7 + 1,8|

Ни в одном из уравнений нет диагонального преобладания. Значит необходимо выполнить преобразования системы.

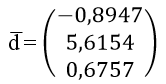
Отнимим от первого уравнения второе и прибавим третье, запишем ответ в первое уравнение; отнимем от второго первое, умноженное на 2, и прибавим третье, запишем ответ во второе уравнение; от второго отнимем третье и запишем ответ в третье уравнение.

Используя следующие преобразования



получим систему имеющую следующий вид:

Теперь вычислим элементы матрицы B и вектора d- :

3. Вычислим норму матрицы B:

||В||1 = max ∑|bij|, j = 1,n.

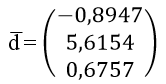
||В||1 = max{0,1053+0,4737; +;+} = max {0,579; 0,3846; 0,89199} = 0,8919 < 1 => процесс продолжается.

4. Вычислим норму вектора D:

||d||1 = max |di|, i=1,n.

||d||1 =5,6154.

5. Выберем в качестве начального приближения вектор d:



6. Вычислим необходимое число итераций для достижения требуемой точности решения системы:

K<= - 1

K<= – 1

K = 9

7. Реализовываем итерационный процесс:

1 итерация:

Новое приближение:

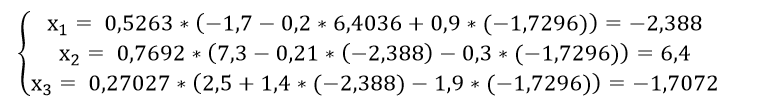
x1 =

2 итерация:

Новое приближение:

x2 =

3 итерация:



Ответ:

x3 =

РЕШЕНИЕ СИСТЕМЫ С ПОМОЩЬЮ ПРОГРАММЫ

МЕТОД ГАУССА-ЗЕЙДЕЛЯ

*Студента Костюка И.Ю.*

Введите точность: 0.001

Введенная система:

3,3\*X1- 2,1\*X2+ 2,8\*X3 = 0.8

4,1\*X1+ 3,7\*X2+ 4.8\*X3 = 5,7

2.7\*X1+1.8\*X2+ 1,1\*X3 = 3,2

Ответ:

X[1] =

X[2] =

X[3] =