Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное
образовательное учреждение высшего образования
Санкт-Петербургский университет аэрокосмического приборостроения

КАФЕДРА № 2

ОТЧЕТ
ЗАЩИЩЕН С ОЦЕНКОЙ
ПРЕПОДАВАТЕЛЬ

доц., канд. техн. наук		С.Л. Козенко
должность, уч. степень, звание	подпись, дата	инициалы, фамилия

ОТЧЕТ О ПРАКТИЧЕСКОЙ РАБОТЕ

Решение СЛАУ

по дисциплине: ВЫЧИСЛИТЕЛЬНАЯ МАТЕМАТИКА

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. № 4134	19.03.2023	Д.В. Шумков
	подпись, дата	инициалы, фамилия

Санкт – Петербург, 2023

Цель работы: а) освоение основных методов решения систем линейных алгебраических уравнений (СЛАУ); б) совершенствование навыков по алгоритмизации и программированию вычислительных задач.

Постановка задачи: Составить схему алгоритма и программу на языке C/C++ решения задачи по теме «Решение СЛАУ». Решить систему линейных уравнений $AX = B$ методом обратной матрицы, где

$$A = \begin{pmatrix} 2 & 4 & 7 & 4 \\ 4 & 1 & 6 & 2 \\ 8 & 3 & 6 & 7 \\ 6 & 3 & 5 & 7 \end{pmatrix}, \quad B = \begin{pmatrix} 2 \\ 0 \\ 3 \\ 1 \end{pmatrix}$$

Математическая часть

Один из методов решения системы линейных алгебраических уравнений, записываемой в матричной форме A × C = B, связан с использованием обратной матрицы A^{-1}.

В этом случае решение системы уравнений получается в виде $C = A^{-1} * B$

Где A^{-1} – матрица, определяемая следующим образом. Пусть A – квадратная матрица размером m×m с ненулевым определителем det A ≠ 0. Тогда существует обратная матрица R = A^{-1}, определяемая условием A × R = E, где E – единичная матрица, размерности m×m все элементы главной диагонали которой равны 1, а все элементы вне этой диагонали равны 0.

$$\mathbf{E} = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 1 \end{bmatrix}.$$

Матрица R – квадратная матрица размером m×m.

$$\mathbf{R} = \begin{bmatrix} r_{11} & r_{12} & \dots & r_{1m} \\ r_{21} & r_{22} & \dots & r_{2m} \\ \dots & \dots & \dots & \dots \\ r_{m1} & r_{m2} & \dots & r_{mm} \end{bmatrix}.$$

Тогда можно записать в матричном виде

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{2m} \\ \dots & \dots & \dots & \dots \\ a_{m1} & a_{m2} & \dots & a_{mm} \end{bmatrix} \times \begin{bmatrix} r_{11} & r_{12} & \dots & r_{1m} \\ r_{21} & r_{22} & \dots & r_{2m} \\ \dots & \dots & \dots & \dots \\ r_{m1} & r_{m2} & \dots & r_{mm} \end{bmatrix} = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 1 \end{bmatrix}.$$

Умножим матрицу А на первый столбец матрицы R. Получим первый столбец матрицы Е, то есть система уравнений будет представлена в виде

$$
\begin{gathered}
a_{11} r_{11} + a_{12} r_{21} + \dots + a_{1m} r_{m1} = 1 \\
a_{21} r_{11} + a_{22} r_{21} + \dots + a_{2m} r_{m1} = 0, \\
\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \\
a_{m1} r_{11} + a_{m2} r_{21} + \dots + a_{mm} r_{m1} = 0
\end{gathered}
$$

где неизвестными являются значения r11 , r21,…, rm1. Решением этой системы будет набор значений r11 , r21,…, rm1, то есть первый столбец матрицы R. Аналогично, умножая матрицу А на второй столбец матрицы R, получим:

$$
\begin{gathered}
a_{11} r_{12} + a_{12} r_{22} + \dots + a_{1m} r_{m2} = 0 \\
a_{21} r_{12} + a_{22} r_{22} + \dots + a_{2m} r_{m2} = 1 \\
\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \\
a_{m1} r_{12} + a_{m2} r_{22} + \dots + a_{mm} r_{m2} = 0.
\end{gathered}
$$

Решением этой системы будет набор значений r12, r22,…, rm2, то есть второй столбец матрицы R. И так далее до m-го столбца матрицы R:

$$
\begin{gathered}
a_{11} r_{1m} + a_{12} r_{2m} + \dots + a_{1m} r_{mm} = 0 \\
a_{21} r_{1m} + a_{22} r_{2m} + \dots + a_{2m} r_{mm} = 0 \\
\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \\
a_{m1} r_{1m} + a_{m2} r_{2m} + \dots + a_{mm} r_{mm} = 1.
\end{gathered}
$$

Решением этой системы будет набор значений r1m, r2m,…, rmm, то есть m– й столбец матрицы R. В результате получим m систем уравнений. Для решения этих систем можно применять любые методы, разработанные для решения систем линейных алгебраических уравнений. Вычислив матрицу R (матрицу А-1), находим матрицу коэффициентов С.

Аналитические расчеты

Для выполнения аналитических расчетов был использовал онлайн калькулятор.

Транспонированная матрица к матрице A имеет вид:

A^T=

2	4	8	6
4	1	3	3
7	6	6	5
4	2	7	7

Вычисляем алгебраические дополнения.

$A^T_{1,1}=(-1)^{1+1}$

1	3	3
6	6	5
2	7	7

$\Delta_{1,1}$=1•(6•7-7•5)-6•(3•7-7•3)+2•(3•5-6•3)=1

$$A^T_{1,2}=(-1)^{1+2}\begin{vmatrix} 4 & 3 & 3 \\ 7 & 6 & 5 \\ 4 & 7 & 7 \end{vmatrix}$$

$\Delta_{1,2}$=-4•(6•7-7•5)-7•(3•7-7•3)+4•(3•5-6•3)=-16

$$A^T_{1,3}=(-1)^{1+3}\begin{vmatrix} 4 & 1 & 3 \\ 7 & 6 & 5 \\ 4 & 2 & 7 \end{vmatrix}$$

$\Delta_{1,3}$=4•(6•7-2•5)-7•(1•7-2•3)+4•(1•5-6•3)=69

$$A^T_{1,4}=(-1)^{1+4}\begin{vmatrix} 4 & 1 & 3 \\ 7 & 6 & 6 \\ 4 & 2 & 7 \end{vmatrix}$$

$\Delta_{1,4}$=-4•(6•7-2•6)-7•(1•7-2•3)+4•(1•6-6•3)=-65

$$A^T_{2,1}=(-1)^{2+1}\begin{vmatrix} 4 & 8 & 6 \\ 6 & 6 & 5 \\ 2 & 7 & 7 \end{vmatrix}$$

$\Delta_{2,1}$=-4•(6•7-7•5)-6•(8•7-7•6)+2•(8•5-6•6)=48

$$A^T_{2,2}=(-1)^{2+2}\begin{vmatrix} 2 & 8 & 6 \\ 7 & 6 & 5 \\ 4 & 7 & 7 \end{vmatrix}$$

$\Delta_{2,2}$=2•(6•7-7•5)-7•(8•7-7•6)+4•(8•5-6•6)=-68

$$A^T_{2,3}=(-1)^{2+3}\begin{vmatrix} 2 & 4 & 6 \\ 7 & 6 & 5 \end{vmatrix}$$

$$\begin{vmatrix} 4 & 2 & 7 \end{vmatrix}$$

$\Delta_{2,3}=-2\cdot(6\cdot7-2\cdot5)-7\cdot(4\cdot7-2\cdot6)+4\cdot(4\cdot5-6\cdot6)=112$

$$A^{T}_{2,4}=(-1)^{2+4}\begin{vmatrix} 2 & 4 & 8 \\ 7 & 6 & 6 \\ 4 & 2 & 7 \end{vmatrix}$$

$\Delta_{2,4}=2\cdot(6\cdot7-2\cdot6)-7\cdot(4\cdot7-2\cdot8)+4\cdot(4\cdot6-6\cdot8)=-120$

$$A^{T}_{3,1}=(-1)^{3+1}\begin{vmatrix} 4 & 8 & 6 \\ 1 & 3 & 3 \\ 2 & 7 & 7 \end{vmatrix}$$

$\Delta_{3,1}=4\cdot(3\cdot7-7\cdot3)-1\cdot(8\cdot7-7\cdot6)+2\cdot(8\cdot3-3\cdot6)=-2$

$$A^{T}_{3,2}=(-1)^{3+2}\begin{vmatrix} 2 & 8 & 6 \\ 4 & 3 & 3 \\ 4 & 7 & 7 \end{vmatrix}$$

$\Delta_{3,2}=-2\cdot(3\cdot7-7\cdot3)-4\cdot(8\cdot7-7\cdot6)+4\cdot(8\cdot3-3\cdot6)=32$

$$A^{T}_{3,3}=(-1)^{3+3}\begin{vmatrix} 2 & 4 & 6 \\ 4 & 1 & 3 \\ 4 & 2 & 7 \end{vmatrix}$$

$\Delta_{3,3}=2\cdot(1\cdot7-2\cdot3)-4\cdot(4\cdot7-2\cdot6)+4\cdot(4\cdot3-1\cdot6)=-38$

$$A^{T}_{3,4}=(-1)^{3+4}\begin{vmatrix} 2 & 4 & 8 \\ 4 & 1 & 3 \\ 4 & 2 & 7 \end{vmatrix}$$

$\Delta_{3,4}=-2\cdot(1\cdot7-2\cdot3)-4\cdot(4\cdot7-2\cdot8)+4\cdot(4\cdot3-1\cdot8)=30$

$$A^{T}_{4,1}=(-1)^{4+1}\begin{vmatrix} 4 & 8 & 6 \end{vmatrix}$$

1	3	3
6	6	5

$\Delta_{4,1}$=-4•(3•5-6•3)-1•(8•5-6•6)+6•(8•3-3•6)=-20

$A^T_{4,2}=(-1)^{4+2}$

2	8	6
4	3	3
7	6	5

$\Delta_{4,2}$=2•(3•5-6•3)-4•(8•5-6•6)+7•(8•3-3•6)=20

$A^T_{4,3}=(-1)^{4+3}$

2	4	6
4	1	3
7	6	5

$\Delta_{4,3}$=-2•(1•5-6•3)-4•(4•5-6•6)+7•(4•3-1•6)=-80

$A^T_{4,4}=(-1)^{4+4}$

2	4	8
4	1	3
7	6	6

$\Delta_{4,4}$=2•(1•6-6•3)-4•(4•6-6•8)+7•(4•3-1•8)=100

Из полученных алгебраических дополнений составим присоединенную матрицу C:

C=

1	-16	69	-65
48	-68	112	-120
-2	32	-38	30
-20	20	-80	100

Вычислим обратную матрицу:

$A^{-1} = \frac{1}{100}$

1	-16	69	-65
48	-68	112	-120

-2	32	-38	30
-20	20	-80	100

Вектор результатов X

$X=A^{-1} \cdot B$

$X = \frac{1}{100}$

1	-16	69	-65
48	-68	112	-120
-2	32	-38	30
-20	20	-80	100

x

2
0
3
1

$X = \frac{1}{100}$

(1*2)+(-16*0)+(69*3)+(-65*1)
(48*2)+(-68*0)+(112*3)+(-120*1)
(-2*2)+(32*0)+(-38*3)+(30*1)
(-20*2)+(20*0)+(-80*3)+(100*1)

$X = \frac{1}{100}$

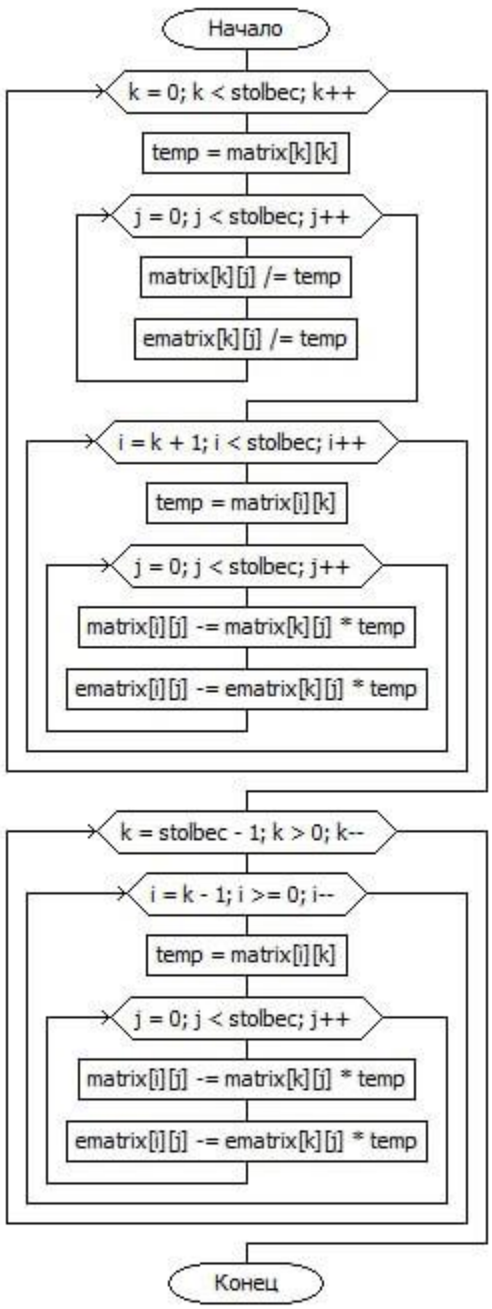
144
312
-88
-180

X^T=(1.44,3.12,-0.88,-1.8)

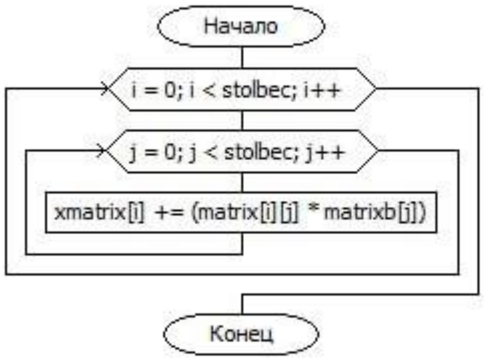
$x_1 = {}^{144} / {}_{100} = 1.44$

$x_2 = {}^{312}/_{100} = 3.12$
$x_3 = {}^{-88}/_{100} = -0.88$
$x_4 = {}^{-180}/_{100} = -1.8$
Проверка.
2•1.44+4•3.12+7•(-0.88)+4•(-1.8)=2
4•1.44+1•3.12+6•(-0.88)+2•(-1.8)=0
8•1.44+3•3.12+6•(-0.88)+7•(-1.8)=3
6•1.44+3•3.12+5•(-0.88)+7•(-1.8)=1

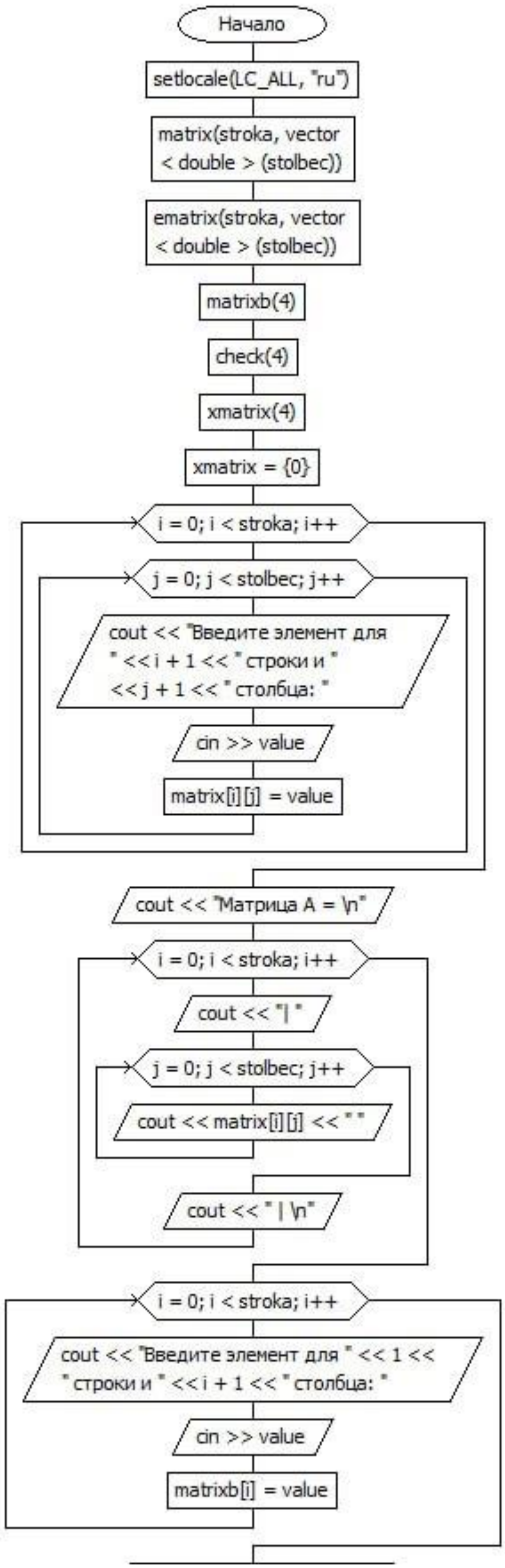
Схема алгоритма
Функция invers

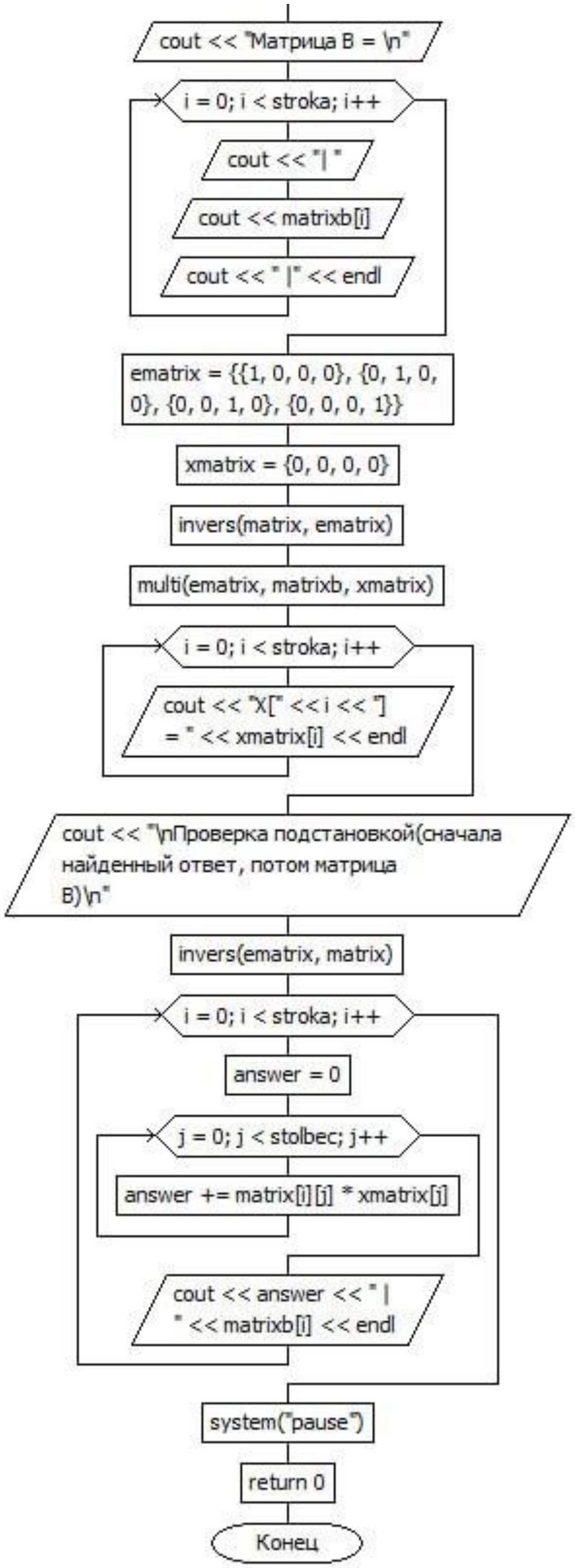


Функция multi



Main


Начало
setlocale(LC_ALL, "ru")
matrix(stroka, vector < double > (stolbec))
ematrix(stroka, vector < double > (stolbec))
matrixb(4)
check(4)
xmatrix(4)
xmatrix = {0}
i = 0; i < stroka; i++
j = 0; j < stolbec; j++
cout << "Введите элемент для " << i + 1 << " строки и " << j + 1 << " столбца: "
cin >> value
matrix[i][j] = value
cout << "Матрица A = \n"
i = 0; i < stroka; i++
cout << "| "
j = 0; j < stolbec; j++
cout << matrix[i][j] << " "
cout << " | \n"
i = 0; i < stroka; i++
cout << "Введите элемент для " << 1 << " строки и " << i + 1 << " столбца: "
cin >> value
matrixb[i] = value


cout << "Матрица B = \n"
i = 0; i < stroka; i++
cout << "| "
cout << matrixb[i]
cout << " |" << endl
ematrix = {{1, 0, 0, 0}, {0, 1, 0, 0}, {0, 0, 1, 0}, {0, 0, 0, 1}}
xmatrix = {0, 0, 0, 0}
invers(matrix, ematrix)
multi(ematrix, matrixb, xmatrix)
i = 0; i < stroka; i++
cout << "X[" << i << "] = " << xmatrix[i] << endl
cout << "\nПроверка подстановкой(сначала найденный ответ, потом матрица B)\n"
invers(ematrix, matrix)
i = 0; i < stroka; i++
answer = 0
j = 0; j < stolbec; j++
answer += matrix[i][j] * xmatrix[j]
cout << answer << " | " << matrixb[i] << endl
system("pause")
return 0
Конец

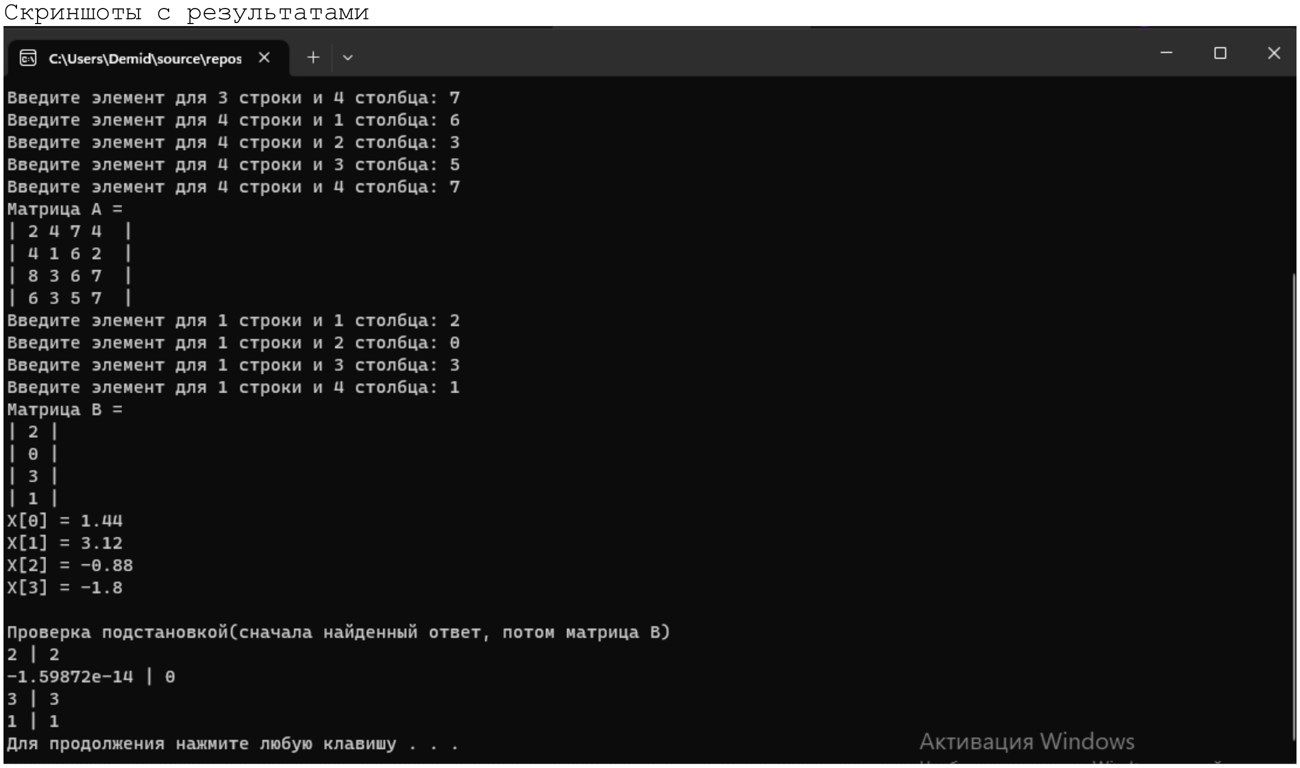
Листинг программы

```
#include <iostream>
#include <vector>
using namespace std;

//функция, вычисляющая обратную матрицу
const double stroka = 4;
const double stolbec = 4;
void invers(vector<vector<double>>& matrix, vector<vector<double>>& ematrix)
{
	double temp;
	for (int k = 0; k < stolbec; k++)
	{
		temp = matrix[k][k];
		for (int j = 0; j < stolbec; j++)
		{
			matrix[k][j] /= temp;
			ematrix[k][j] /= temp;
		}

		for (int i = k + 1; i < stolbec; i++)
		{
			temp = matrix[i][k];
			for (int j = 0; j < stolbec; j++)
			{
				matrix[i][j] -= matrix[k][j] * temp;
				ematrix[i][j] -= ematrix[k][j] * temp;
			}
		}
	}

	for (int k = stolbec - 1; k > 0; k--)
	{
		for (int i = k - 1; i >= 0; i--)
		{
			temp = matrix[i][k];

			for (int j = 0; j < stolbec; j++)
			{
				matrix[i][j] -= matrix[k][j] * temp;
				ematrix[i][j] -= ematrix[k][j] * temp;
			}
		}
	}
}

// функция, перемножающая матрицу А на вектор В
void multi(vector<vector<double>>& matrix, vector<double>& matrixb, vector<double>&
xmatrix)
{
	for (int i = 0; i < stolbec; i++)
		for (int j = 0; j < stolbec; j++)
			xmatrix[i] += (matrix[i][j] * matrixb[j]);
}

int main()
{
	setlocale(LC_ALL, "ru");
	double value;
	vector<vector <double>> matrix(stroka, vector<double>(stolbec));
	vector<vector <double>> ematrix(stroka, vector<double>(stolbec));
	vector <double> matrixb(4);
```

```
	vector <double> check(4);
	vector <double> xmatrix(4);
	xmatrix = { 0 };
	for (int i = 0; i < stroka; i++)
	{
		for (int j = 0; j < stolbec; j++)
		{
			cout << "Введите элемент для " << i + 1 << " строки и " << j + 1
<< " столбца: ";
			cin >> value;
			matrix[i][j] = value;
		}
	}

	cout << "Матрица A = \n";
	for (int i = 0; i < stroka; i++)
	{
		cout << "| ";
		for (int j = 0; j < stolbec; j++)
		{
			cout << matrix[i][j] << " ";
		}
		cout << " | \n";
	}
	for (int i = 0; i < stroka; i++)
	{

		cout << "Введите элемент для " << 1 << " строки и " << i + 1 << "
столбца: ";
		cin >> value;
		matrixb[i] = value;

	}
	cout << "Матрица B = \n";
	for (int i = 0; i < stroka; i++)
	{
		cout << "| ";
		cout << matrixb[i];
		cout << " |" << endl;
	}

	ematrix = {
				{ 1, 0, 0, 0 },
				{ 0, 1, 0, 0 },
				{ 0, 0, 1, 0 },
				{ 0, 0, 0, 1 }
	}; // единичная матрица
	xmatrix = { 0,0,0,0 };
	invers(matrix, ematrix); // находим обратную матрицу A^-1
	multi(ematrix, matrixb, xmatrix); // перемножаем обратную матрицу A^-1 на
вектор B
	for (int i = 0; i < stroka; i++)
		cout << "X[" << i << "] = " << xmatrix[i] << endl;
	cout << "\nПроверка подстановкой(сначала найденный ответ, потом матрица B)\n";
	invers(ematrix, matrix);
	for (int i = 0; i < stroka; i++)
	{
		double answer = 0;
		for (int j = 0; j < stolbec; j++)
		{
			answer += matrix[i][j] * xmatrix[j];
		}
		cout << answer << " | " << matrixb[i] << endl;
```

```
	}
	system("pause");
	return 0;
}
```

Скриншоты с результатами



Сравнение результатов программных и аналитических расчетов

Результаты аналитического расчета x1 = 1,44; x2 = 3,12; x3 = -0,88; x4 = -1,8. Результаты программного расчета x1 = 1,44; x2 = 3,12; x3 = -0,88; x4 = -1,8. Исходя из результатов, полученных при помощи онлайн калькулятора, и данных, выводимых программой, можно увидеть, что результаты совпадают. Отсюда следует, что выводимые программой данные достоверны.

Выводы

Так как аналитические расчеты и программные результаты совпадают, можно сделать вывод, что программа работает корректно и справляется с поставленной задачей, а именно решает систему линейных алгебраических уравнений методом обратной матрицы.
В ходе выполнения лабораторной работы были получены знания по решению СЛАУ методом обратной матрицы с помощью языка программирования C++. Данная лабораторная работа способствовала совершенствованию навыков по алгоритмизации и программированию вычислительных задач.