МИНЕСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение высшего образования

«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

КАФЕДРА ПРИКЛАДНОЙ МАТЕМАТИКИ

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| доц., канд. техн. наук |  |  |  | О. М. Косогоров |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ |
| Численные методы Решения СЛАУ. |
| по курсу: ВЫЧИСЛИТЕЛЬНАЯ МАТЕМАТИКА |
|  |
|  |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. № | М023 |  |  |  | Д.А.Трегуб |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург 2021

**1)Задание:** Решить две СЛАУ(с применением метода Гаусса) и вычислить два матричных выражения со своим вариантом задания.

Вариант № 9:

1) Решить системы линейных уравнений   и вычислить значения квадратичной формы , где

, , 

2)Вычислить  ,  
где x, y – векторы из n компонентов, причём n = 4 и .

**2)Код на C++:**

#include <iostream>

#include <cassert>

using namespace std;

class SLAU {

private:

double\*\* a, \* y, \* x;

int n;

public:

SLAU(int w) {

n = w;

a = new double\* [n];

for (int i = 0; i < n; i++)

a[i] = new double[n];

y = new double[n];

x = new double[n];

}

~SLAU() {

for (int i = 0; i < n; i++)

delete[] a[i];

delete[] a;

delete[] y;

delete[] x;

}

void enter() {

cout << "Введите коофиценты: " << endl;

for (int i = 0; i < n; i++)

{

for (int j = 0; j < n; j++)

cin >> a[i][j];

}

cout << "Введите правую часть уравнения: " << endl;

for (int i = 0; i < n; i++)

cin >> y[i];

}

void enter\_vect() {

cout << "Введите значения вектора: " << endl;

for (int j = 0; j < n; j++)

cin >> a[0][j];

for (int i = 1; i < n; i++) {

for (int j = 0; j < n; j++)

a[i][j] = 0;

}

}

void system\_equat()

{

cout << "Система уравнений: " << endl;

for (int i = 0; i < n; i++)

{

for (int j = 0; j < n; j++)

{

cout << a[i][j] << "\*x" << j;

if (j < n - 1)

cout << " + ";

}

cout << " = " << y[i] << endl;

}

return;

}

void system\_result() {

cout << "Ответ: " << endl;

for (int i = 0; i < n; i++)

cout << "x[" << i << "]=" << x[i] << endl;

cin.get();

}

void transp()

{

double\*\* at = new double\* [n];

for (int i = 0; i < n; i++)

at[i] = new double[n];

for (int i = 0; i < n; i++) {

for (int j = 0; j < n; j++)

at[j][i] = a[i][j];

}

for (int i = 0; i < n; i++) {

for (int j = 0; j < n; j++)

a[i][j] = at[i][j];

}

for (int i = 0; i < n; i++)

delete[] at[i];

delete[] at;

return;

}

double\*\* get\_matr() {return a;}

void set\_matr(double\*\* c) {

for (int i = 0; i < n; i++) {

for (int j = 0; j < n; j++)

a[i][j] = c[i][j];

}

}

void mpn(double\*\* b) { // умножение матрицы

double\*\* c = new double\* [n];

for (int i = 0; i < n; i++)

{

c[i] = new double[n];

for (int j = 0; j < n; j++)

{

c[i][j] = 0;

for (int k = 0; k < n; k++)

c[i][j] += a[i][k] \* b[k][j];

}

}

for (int i = 0; i < n; i++) {

for (int j = 0; j < n; j++) {

if (c[i][j] <= 0.00001)

c[i][j] = 0;

a[i][j] = c[i][j];

}

}

for (int i = 0; i < n; i++)

delete[] c[i];

delete[] c;

}

void gauss()

{

double max;

int k, index;

const double eps = 0.000000001; // точность

k = 0;

while (k < n)

{

// Поиск строки с максимальным a[i][k]

max = abs(a[k][k]);

index = k;

for (int i = k + 1; i < n; i++)

{

if (abs(a[i][k]) > max)

{

max = abs(a[i][k]);

index = i;

}

}

// Перестановка строк

for (int j = 0; j < n; j++)

{

double temp = a[k][j];

a[k][j] = a[index][j];

a[index][j] = temp;

}

double temp = y[k];

y[k] = y[index];

y[index] = temp;

// Нормализация уравнений

for (int i = k; i < n; i++)

{

double temp = a[i][k];

if (abs(temp) < eps) continue; // для нулевого коэффициента пропустить

for (int j = 0; j < n; j++)

a[i][j] = a[i][j] / temp;

y[i] = y[i] / temp;

if (i == k) continue; // уравнение не вычитать само из себя

for (int j = 0; j < n; j++)

a[i][j] = a[i][j] - a[k][j];

y[i] = y[i] - y[k];

}

k++;

}

// обратная подстановка

for (k = n - 1; k >= 0; k--)

{

x[k] = y[k];

for (int i = 0; i < k; i++)

y[i] = y[i] - a[i][k] \* x[k];

}

}

void out() {

for (int i = 0; i < n; i++) {

for (int j = 0; j < n; j++) {

cout << a[i][j] << " ";

}

cout << endl;

}

cout << "Матрица:" << endl;

}

double front() {

return a[0][0];

};

double sum(double\*\* c, double\*\* v) {

double sum = 0;

for (int i = 0; i < n; i++) {

sum += c[0][i] \* v[0][i];

}

return sum;

}

double sum(double\*\* c) {

double sum = 0;

for (int i = 0; i < n; i++) {

sum += c[0][i];

}

return sum;

}

};

int main()

{

system("chcp 1251");

system("cls");

cout << "Первое уравнение:" << endl;

SLAU q = SLAU(4);

q.enter();

q.system\_equat();

q.gauss();

q.system\_result();

cout << endl << "Второе уравнение:" << endl;

SLAU z = SLAU(4);

SLAU t = SLAU(4);

SLAU r = SLAU(4);

z.enter();

t.set\_matr(z.get\_matr());

t.transp();

t.mpn(z.get\_matr());

r.set\_matr(z.get\_matr());

r.transp();

t.mpn(r.get\_matr());

z.set\_matr(t.get\_matr());

z.system\_equat();

z.gauss();

z.system\_result();

cout << endl << "Значение квадратичной формы: " << endl;

SLAU a = SLAU(4);

SLAU a\_tr = SLAU(4);

SLAU y = SLAU(4);

SLAU y\_tr = SLAU(4);

a.enter();

a\_tr.set\_matr(a.get\_matr());

a\_tr.transp();

y\_tr.enter\_vect();

y.set\_matr(y\_tr.get\_matr());

y.transp();

y\_tr.mpn(a.get\_matr());

y\_tr.mpn(a\_tr.get\_matr());

y\_tr.mpn(a.get\_matr());

y\_tr.mpn(a\_tr.get\_matr());

y\_tr.mpn(y.get\_matr());

cout << "Ответ: " << fixed << y\_tr.front();

cout << endl << endl << "Вычисление выражения:" << endl;

SLAU g = SLAU(4);

SLAU h = SLAU(4);

g.enter\_vect();

h.enter\_vect();

cout << "Ответ:" << (g.sum(g.get\_matr(), g.get\_matr()) + 5 \* g.sum(g.get\_matr(), h.get\_matr())) \* (1 + g.sum(g.get\_matr()) + g.sum(h.get\_matr()) - 3 + g.sum(g.get\_matr(), g.get\_matr()));

return 0;

}

/\*

7 6 2 7

4 9 5 5

2 3 4 9

1 5 6 9

3 2 0 2

7 6 2 7

4 9 5 5

2 3 4 9

1 5 6 9

3 2 0 2

7 6 2 7

4 9 5 5

2 3 4 9

1 5 6 9

0 0 0 0

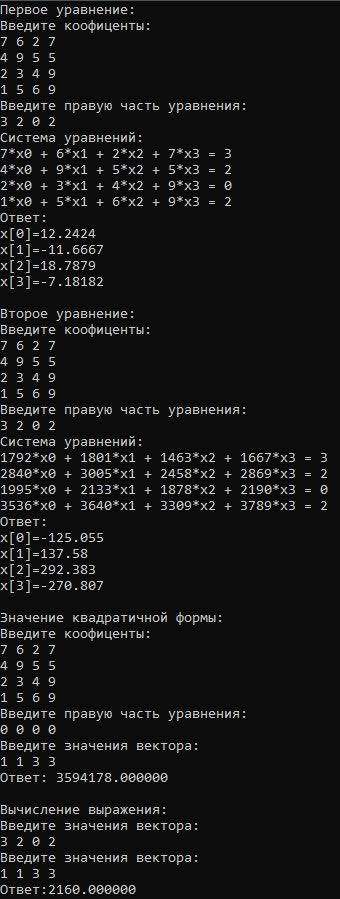
1 1 3 3

3 2 0 2

1 1 3 3

\*/

**3)Результат работы программы с++:**



**4)Код проверки СЛАУ на матлабе:**

A=[7 6 2 7; 4 9 5 5; 2 3 4 9; 1 4 6 9]; %первое уравнение

B=[3; 2; 0; 2];

x=A\B

A=[7 6 2 7; 4 9 5 5; 2 3 4 9; 1 4 6 9]; %Второе уравнение

B=[3; 2; 0; 2];

G=A';

С = G\*A\*G;

x=C\B;

**5)Вывод:**

Выполняя данную лабораторную работу я решил 2 СЛАУ с применением метода Гаусса, реализовав его на С++, и проверил их с помощью Matlab, а также 2 матричных выражения, также реализовав операции транспонирования и умножения матриц на С++.