МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
АДЫГЕЙСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Инженерно-физический факультет
Кафедра автоматизированных систем обработки информации и
управления

Отчет по практике

Решение системы линейных алгебраических уравнений методом Гаусса-Жордана.

1 курс, группа 1ИВТ

Выполнил:
________________ Д. Э. Небольсин
«____» __________ 2023 г.

Руководитель:
________________ С. В. Теплоухов
«____» __________ 2023 г.

Майкоп, 2023 г.

Содержание

1. Теория

1.1. Техническое задание

Задание:

Решение системы линейных алгебраических уравнений методом Гаусса-Жордана.

1.2. Теоретическая часть

Метод Гаусса - Жордана (метод полного исключения неизвестных) - метод, который используется для решения квадратных систем линейных алгебраических уравнений, нахождения обратной матрицы, нахождения координат вектора в заданном базисе или отыскания ранга матрицы. Метод является модификацией метода Гаусса.

Алгоритм:

1) Выбирают первый слева столбец матрицы, в котором есть хоть одно отличное от нуля значение.

2) Если самое верхнее число в этом столбце ноль, то меняют всю первую строку матрицы с другой строкой матрицы, где в этой колонке нет нуля.

3) Все элементы первой строки делят на верхний элемент выбранного столбца.

4) Из оставшихся строк вычитают первую строку, умноженную на первый элемент соответствующей строки, с целью получить первым элементом каждой строки (кроме первой) ноль.

5) Далее проводят такую же процедуру с матрицей, получающейся из исходной матрицы после вычёркивания первой строки и первого столбца.

6) После повторения этой процедуры n-1 раз получают верхнюю треугольную матрицу.

7) Вычитают из предпоследней строки последнюю строку, умноженную на соответствующий коэффициент, с тем, чтобы в предпоследней строке осталась только 1 на главной диагонали.

8) Повторяют предыдущий шаг для последующих строк. В итоге получают единичную матрицу и решение на месте свободного вектора (с ним необходимо проводить все те же преобразования).

2. Ход работы

2.1. Код приложения

```
#include <iostream>
#include <conio.h>
#include <stdio.h>
#include <string>
#include <cmath>

using namespace std;

double getValue(int s, int c, int n, double *m)
{
    return m[s * (n + 1) + c];
}

void setValue(double v, int s, int c, int n, double *m)
{
    m[s * (n + 1) + c] = v;
}

void printMatrix(int n, double *m)
{
    for (int s = 0; s < n; s++)
    {
        for (int c = 0; c < n + 1; c++)
        {
            cout << getValue(s, c, n, &m[0]) << "  ";
        }
        cout << "\n";
    }
    cout << "\n";
}

void setLine(int s, int n, double *m)
{
    double tmp;

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

```
        if (getValue(i, s, n, m) != 0)
        {
            if (i != 0)
            {
                for (int j = s; j < n + 1; j++)
                {
                    tmp = getValue(s, j, n, m);
                    setValue(getValue(i, j, n, m), s, j, n, m);
                    setValue(tmp, i, j, n, m);
                }
            }
            break;
        }
    }
}

void makeLine(int s, int n, double *m)
{
    double k = getValue(s, s, n, m);
    for (int i = s; i < n + 1; i++)
    {
        setValue(getValue(s, i, n, m) / k, s, i, n, m);
    }
}

void setZeroLeftBottom(int s, int n, double *m)
{
    double tmp, tmp1, k;
    for (int i = s + 1; i < n; i++)
    {
        k = getValue(i, s, n, m);
        for (int j = s; j < n + 1; j++)
        {
            tmp = getValue(s, j, n, m);
            tmp1 = getValue(i, j, n, m);
            setValue(tmp * (-k) + tmp1, i, j, n, m);
        }
    }
}
```

```
void setZeroRightTop(int s, int n, double *m)
{
    double tmp, tmp1, k;
    for (int i = s - 1; i >= 0; i--)
    {
        k = getValue(i, s, n, m);
        for (int j = s; j < n + 1; j++)
        {
            tmp = getValue(s, j, n, m);
            tmp1 = getValue(i, j, n, m);
            setValue(tmp * (-k) + tmp1, i, j, n, m);
        }
    }
}

bool isNull (float value)
{
    if (std::abs(value) < 0.0000001)
    {
        return true;
    }
    return false;
}

int main(void)
{
    int n, s, c;

    cout << "Размерность = ";
    cin >> n;

    double matrix[n][n + 1];

    for (s = 0; s < n; s++)
    {
        cout << "\n";
        for (c = 0; c < n + 1; c++)
        {
            cout << "a[" << s << "][" << c << "] = ";
            cin >> matrix[s][c];
```

```
    }
}
cout << "\n";
printMatrix(n, &matrix[0][0]);

cout << "Левый угол\n";

for (int s = 0; s < n; s++)
{
    setLine(s, n, &matrix[0][0]);

    if (isNull(getValue(s, s, n, &matrix[0][0])))
    {
        if (isNull(getValue(s, n, n, &matrix[0][0])))
        {
            cout << "Бесконечное количество решений";
        }
        else
        {
            cout << "Решений нет";
        }
        return 0;
    }

    makeLine(s, n, &matrix[0][0]);

    setZeroLeftBottom(s, n, &matrix[0][0]);

    printMatrix(n, &matrix[0][0]);
}

cout << "Правый угол\n";

for (int s = n - 1; s >= 0; s--)
{
    setZeroRightTop(s, n, &matrix[0][0]);
    printMatrix(n, &matrix[0][0]);
}

cout << "Решение уравнения\n";
```

```
    for (int i = 0; i < n; i++)
    {
        cout    << "X["<< i+1 << "] = "
                << getValue(i, n, n, &matrix[0][0])
                << "\n";
    }
}
```

2.2. Работа программы

2.2.1. Пример системы уравнений с решением

$$\begin{cases} a + b + c = 0 \\ 4a + 2b + c = 1 \\ 9a + 3b + c = 3 \end{cases}$$

Рис.1 Пример системы линейных алгебраических уравнений

```
Размерность = 3

a[0][0] = 1
a[0][1] = 1
a[0][2] = 1
a[0][3] = 0

a[1][0] = 4
a[1][1] = 2
a[1][2] = 1
a[1][3] = 1

a[2][0] = 9
a[2][1] = 3
a[2][2] = 1
a[2][3] = 3
```

Рис.2 Пример входных данных

```
Решение уравнения
X[1] = 0.5
X[2] = -0.5
X[3] = 0
```

Рис.3 Пример вывода данных

2.2.2. Пример системы уравнений без решения

$$\begin{cases} 7x_1 - 2x_2 - x_3 = 2 \\ 6x_1 - 4x_2 - 5x_3 = 3 \\ x_1 + 2x_2 + 4x_3 = 5 \end{cases}$$

Рис.4 Пример системы линейных алгебраических уравнений

```
Размерность = 3

a[0][0] = 7
a[0][1] = -2
a[0][2] = -1
a[0][3] = 2

a[1][0] = 6
a[1][1] = -4
a[1][2] = -5
a[1][3] = 3

a[2][0] = 1
a[2][1] = 2
a[2][2] = 4
a[2][3] = 5
```

Рис.5 Пример входных данных

```
Решений нет
```

Рис.6 Пример вывода данных

2.2.3. Пример системы уравнений с бесконечным количеством решений

$$\begin{cases} x_1 + x_2 - x_3 = 4 \\ 3x_1 + 2x_2 - 5x_3 = 7 \\ 3x_1 + x_2 - 7x_3 = 2 \end{cases}$$

Рис.7 Пример системы линейных алгебраических уравнений

```
Размерность = 3

a[0][0] = 1
a[0][1] = 1
a[0][2] = -1
a[0][3] = 4

a[1][0] = 3
a[1][1] = 2
a[1][2] = -5
a[1][3] = 7

a[2][0] = 3
a[2][1] = 1
a[2][2] = -7
a[2][3] = 2

1  1  -1  4
3  2  -5  7
3  1  -7  2
```

Рис.8 Пример входных данных

```
Бесконечное количество решений
```

Рис.9 Пример вывода данных