

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*

2 курс, группа 2УТС

Выполнил:

_____ А. А. Мугу
«___» _____ 2021 г.

Руководитель:

_____ С. В. Теплоухов
«___» _____ 2021 г.

Майкоп, 2021 г.

1. Введение

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.

2. Ход работы

2.1. Код приложения

```
#include <math.h>
#include <iostream>
using namespace std;

//вариант 2
class Gauss
{
float a[50][50];
int n;
public:
void accept()
{
cout << "Enter no. of variables: ";
cin >> n;
for (int i = 0; i < n; i++)
{
for (int j = 0; j < n + 1; j++)
{
if (j == n)
cout << "Constant no." << i + 1 << " = ";
else
cout << "a[" << i + 1 << "][" << j + 1 << "] = ";
cin >> a[i][j];
}
}
}
void display()
{
for (int i = 0; i < n; i++)
{
cout << "\n";
```

```

for (int j = 0; j < n + 1; j++)
{
    if (j == n)
        cout << " ";
    cout << a[i][j] << "\t";
}
}
}

void gauss()//converting augmented matrix to row echelon form
{
    float temp;//Line 1
    for (int i = 0; i < n; i++)
    {
        for (int j = i + 1; j < n; j++)
        {
            temp = a[j][i] / a[i][i];//Line 2
            for (int k = i; k < n + 1; k++)
            {
                a[j][k] -= temp * a[i][k];//Line 3
                //a[j][k]-=a[j][i]*a[i][k]/a[i][i];//Line 4
            }
        }
    }
}

void EnterJordan()//converting to reduced row echelon form
{
    float temp;
    for (int i = n - 1; i >= 0; i--)
    {
        for (int j = i - 1; j >= 0; j--)
        {
            temp = a[j][i] / a[i][i];
            for (int k = n; k >= i; k--)
            {
                a[j][k] -= temp * a[i][k];
            }
        }
    }
}

float *x = new float [n];
for (int i = 0; i < n; i++)//making leading coefficients zero

```

```

x[i] = 0;
for (int i = 0; i < n; i++)
{
for (int j = 0; j < n + 1; j++)
{
if (x[i] == 0 && j != n)
x[i] = a[i][j];
if (x[i] != 0)
a[i][j] /= x[i];
}
}
delete[]x;
}
void credits()
{
for (int i = 0; i < n; i++)
{
cout << "\nx" << i + 1 << " = " << a[i][n] << endl;
}
}

};

int main()
{
Gauss obj;
obj.accept();
cout << "\n\nAugmented matrix: \n\n\n";
obj.display();
obj.gauss();
cout << "\n\nRow Echelon form: \n\n\n";
obj.display();
obj.EnterJordan();
cout << "\n\nReduced row echelon form:\n\n\n";
obj.display();
cout << "\n\nSolution: \n\n\n";
obj.credits();
return 0;
}

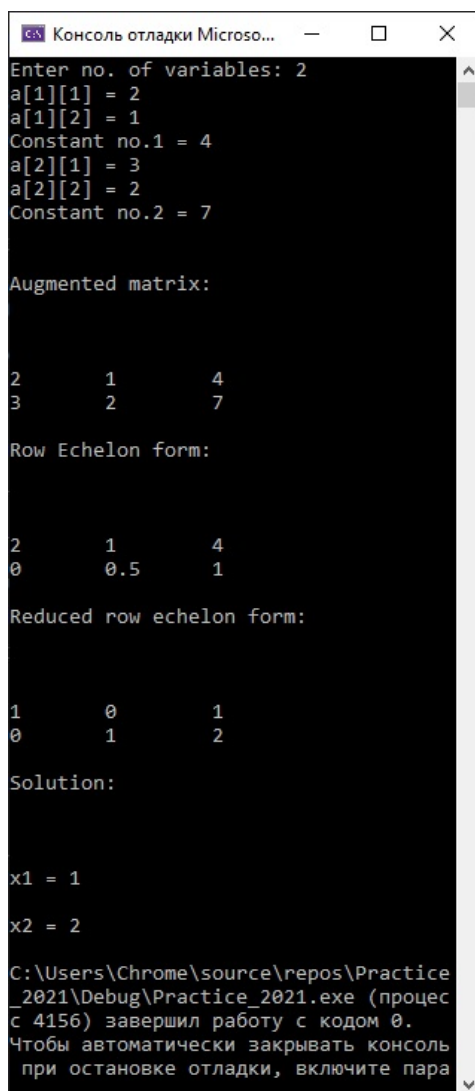
```

2.2. 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).

3. Пример работы программы



```
Консоль отладки Microso...
Enter no. of variables: 2
a[1][1] = 2
a[1][2] = 1
Constant no.1 = 4
a[2][1] = 3
a[2][2] = 2
Constant no.2 = 7

Augmented matrix:

2      1      4
3      2      7

Row Echelon form:

2      1      4
0      0.5    1

Reduced row echelon form:

1      0      1
0      1      2

Solution:

x1 = 1
x2 = 2

C:\Users\Chrome\source\repos\Practice_2021\Debug\Practice_2021.exe (процес
с 4156) завершил работу с кодом 0.
Чтобы автоматически закрывать консоль
при остановке отладки, включите пара
```

Рис. 1. Пример вычислений

Список литературы

- [1] Чакон С. Штрауб Б. Git для профессионального программиста 2016 г.
- [2] Кнут Д.Э. Всё про T_EX. — Москва: Изд. Вильямс, 2003 г. 550 с.
- [3] Львовский С.М. Набор и верстка в системе L^AT_EX. — 3-е издание, исправленное и дополненное, 2003 г.
- [4] Воронцов К.В. L^AT_EX в примерах. 2005 г.