

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

Программная реализация ранга матрицы.
Вариант 6

1 курс, группа 1ИВТ1-2

Выполнил:

_____ Н. С. Сергеев
«___» _____ 2023 г.

Руководитель:

_____ С. В. Теплоухов
«___» _____ 2023 г.

Майкоп, 2023 г.

1. Введение

1.1. Формулировка цели

Целью данной работы является написание программы для нахождения ранга матрицы.

1.1.1. Теория

Нахождение ранга матрицы способом элементарных преобразований (методом Гаусса). Под элементарными преобразованиями матрицы понимаются следующие операции:

- 1) умножение на число, отличное от нуля;
- 2) прибавление к элементам какой-либо строки или какого-либо столбца;
- 3) перемена местами двух строк или столбцов матрицы;
- 4) удаление "нулевых" строк, то есть таких, все элементы которых равны нулю;
- 5) удаление всех пропорциональных строк, кроме одной.

Для любой матрицы A всегда можно прийти к такой матрице B , вычисление ранга которой не представляет затруднений. Для этого следует добиться, чтобы матрица B была трапецевидной. Тогда ранг полученной матрицы будет равен числу строк в ней кроме строк, полностью состоящих из нулей.

Ступенчатую матрицу называют трапецевидной или трапецеидальной, если для ведущих элементов $a_{1k_1}, a_{2k_2}, \dots, a_{rk_r}$ выполнены условия $k_1=1, k_2=2, \dots, k_r=r$, т.е. ведущими являются диагональные элементы. В общем виде трапецевидную матрицу можно записать так:

$$A_{m \times n} = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1r} & \dots & a_{1n} \\ 0 & a_{22} & \dots & a_{2r} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & a_{rr} & \dots & a_{rn} \\ 0 & 0 & \dots & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 0 & \dots & 0 \end{pmatrix}$$

Рис. 1. Трапецевидная матрица

2. Ход работы

2.1. Код выполненной программы

```
#include <iostream>
#include <vector>
#include <stdlib.h>
#include <Windows.h>
#include <cmath>
using namespace std;

int main()
{
    SetConsoleCP(1251);
    SetConsoleOutputCP(1251);
    int height, width, sum = 0, step = 0, sort1, sort2, rank;
    int i, j, k, p, e;
    double tempmath, eps = 0.00001;
    cout << "Введите количество строк матрицы: ";
    cin >> height;
    cout << "Введите количество столбцов матрицы: ";
    cin >> width;
    if (height <= 0 || width <= 0)
    {
        cout << "Ошибка. Неверные параметры матрицы." << endl;
        return 0;
    }
    rank = height;
    vector <vector <double>> matrix;

    for (i = 0; i < height; i++)
    {
        vector <double> temp;
        for (j = 0; j < width; j++)
        {
            cout << "Введите элемент матрицы (" << i + 1 << ", " << j + 1 << "): ";
            cin >> e;
            temp.push_back(e);
        }
        matrix.push_back(temp);
    }
    cout << "\n";
```

```

        cout << "Заданная матрица: ";
for (i = 0; i < height; i++)
{
for (j = 0; j < width; j++)
{
cout << matrix[i][j] << "\t";
}
cout << endl;
}

if (width > height - 1)
{
for (k = 0; k < height - 1; k++)
{
j = k;
for (sort1 = k; sort1 < height; sort1++) {
for (sort2 = k; sort2 < height - 1; sort2++) {
if (abs(matrix[sort2][j]) < abs(matrix[sort2 + 1][j]))
{
for (j = 0; j < width; j++) swap(matrix[sort2][j], matrix[sort2 + 1][j]);
}
j = k;
}
}
for (i = k + 1; i < height; i++)
{
j = k;
tempmath = matrix[i][j] / matrix[i - 1 - step][j];
if (matrix[i][j] == 0)
{
step++;
continue;
}
else
{
for (j = k; j < width; j++)
{
matrix[i - step - 1][j] = matrix[i - step - 1][j] * tempmath;
matrix[i][j] = matrix[i][j] - matrix[i - step - 1][j];
}
step++;
}
}
step = 0;

```

```

for (p = 0; p < width; p++)
if (matrix[k + 1][p] == 0) sum++;
if (sum == width)
{
for (p = 0; p < width; p++)
{
swap(matrix[height - 1][p], matrix[k + 1][p]);
}
}
sum = 0;
}
}

else
{
for (k = 0; k < width; k++)
{
j = k;
for (sort1 = k; sort1 < height; sort1++) {
for (sort2 = k; sort2 < height - 1; sort2++) {
if (matrix[sort2][j] < matrix[sort2 + 1][j])
{
for (j = 0; j < width; j++) swap(matrix[sort2][j], matrix[sort2 + 1][j]);
}
j = k;
}
}
for (i = k + 1; i < height; i++)
{
j = k;
tempmath = matrix[i][j] / matrix[i - 1 - step][j];
if (matrix[i][j] == 0)
{
step++;
continue;
}
else
{
for (j = k; j < width; j++)
{
matrix[i - step - 1][j] = matrix[i - step - 1][j] * tempmath;
matrix[i][j] = matrix[i][j] - matrix[i - step - 1][j];
}
step++;
}
}
}

```

```

}
}
step = 0;
for (p = 0; p < width; p++)
if (matrix[k + 1][p] == 0) sum++;
if (sum == width)
{
for (p = 0; p < width; p++)
{
swap(matrix[height - 1][p], matrix[k + 1][p]);
}
}
sum = 0;
}
}
for (i = 0; i < height; i++)
{
for (j = 0; j < width; j++)
{
if (abs(matrix[i][j]) < eps) matrix[i][j] = 0;
}
}
cout << "\nПриведенная матрица:" << endl;

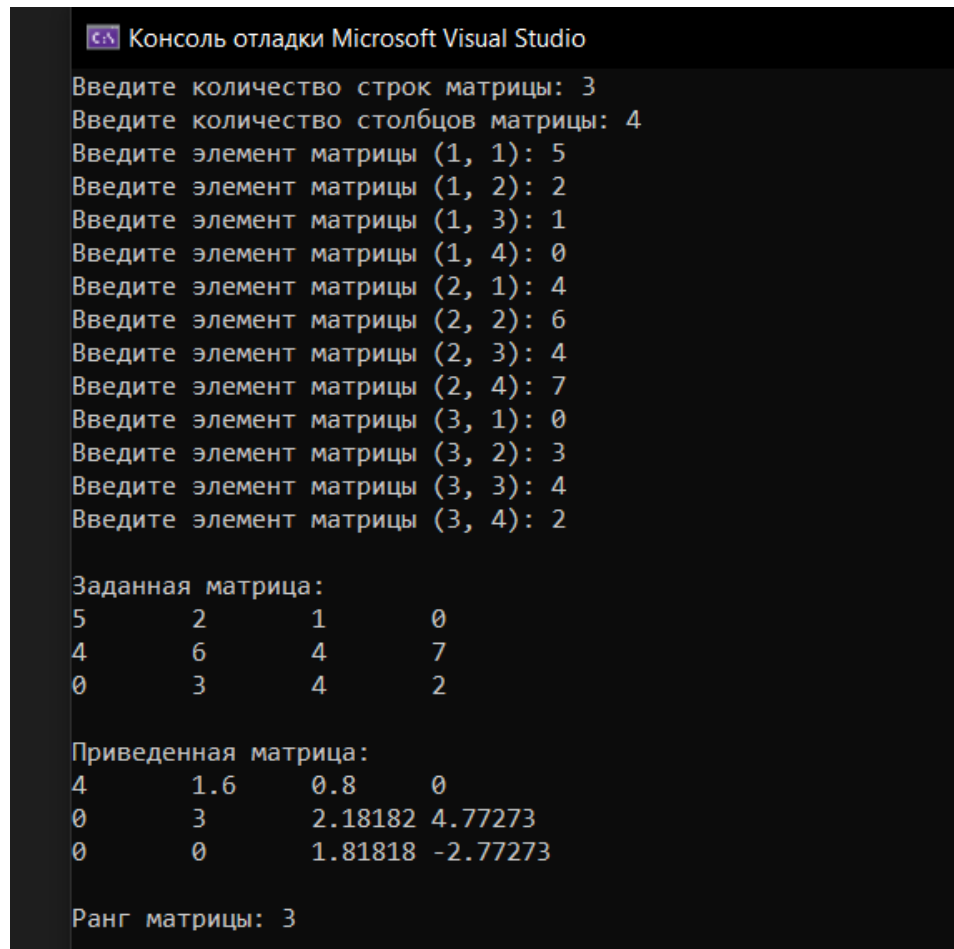
for (i = 0; i < height; i++)
{
for (j = 0; j < width; j++)
{
cout << matrix[i][j] << "\t";
}
cout << endl;
}

for (i = 0; i < height; i++)
{
for (j = 0; j < width; j++)
{
if (matrix[i][j] == 0) sum++;
}
if (sum == width) rank--;
sum = 0;
}

cout << "\nРанг матрицы: " << rank;

```

```
return 0;  
}
```



```
Консоль отладки Microsoft Visual Studio  
Введите количество строк матрицы: 3  
Введите количество столбцов матрицы: 4  
Введите элемент матрицы (1, 1): 5  
Введите элемент матрицы (1, 2): 2  
Введите элемент матрицы (1, 3): 1  
Введите элемент матрицы (1, 4): 0  
Введите элемент матрицы (2, 1): 4  
Введите элемент матрицы (2, 2): 6  
Введите элемент матрицы (2, 3): 4  
Введите элемент матрицы (2, 4): 7  
Введите элемент матрицы (3, 1): 0  
Введите элемент матрицы (3, 2): 3  
Введите элемент матрицы (3, 3): 4  
Введите элемент матрицы (3, 4): 2  
  
Заданная матрица:  
5      2      1      0  
4      6      4      7  
0      3      4      2  
  
Приведенная матрица:  
4      1.6    0.8    0  
0      3      2.18182 4.77273  
0      0      1.81818 -2.77273  
  
Ранг матрицы: 3
```

Рис. 2. Результат работы