



**МИНОБРНАУКИ РОССИИ**

**федеральное государственное бюджетное образовательное учреждение**

**высшего образования**

**«Московский государственный технологический университет**

**«СТАНКИН»**

**(ФГБОУ ВО «МГТУ «СТАНКИН»)**

---

**Институт  
информационных  
технологий**

**Кафедра  
информационных систем**

**Основная образовательная программа 09.03.02**

**«Информационные системы и технологии»**

**Отчет по дисциплине «Структурное программирование»**

**по лабораторной работе № 1**

**по теме: «Обработка двумерных массивов»**

**Студент  
группы ИДБ-21-06**

**Музафаров К.Р.**

**Преподаватель**

**Репин С.В.**

**Москва, 2022 г.**

## **ЗАДАНИЕ**

### **Описание**

Все массивы – динамические. Размерность массивов  $n$  строк и  $m$  столбцов. В 1 задании: вывод программы кроме консоли вывести еще и в файл, вынести логику из основной функцию программы (main) в отдельные функции. Название функции в коде должно быть с большой буквы и должно отображать то, что данная функция выполняет: например, функция CreateArray создает новый массив. Код всех программ оформляется в определенном стиле (объясняется устно на занятии).

## **ЗАДАНИЕ 1**

### **Описание**

Найти среднее арифметическое неотрицательных элементов матрицы, а также подсчитать, сколько таких элементов в каждой отдельно взятой строке матрицы.

## **ЗАДАНИЕ 2**

### **Описание**

Получить массив  $X_1, X_2, \dots, X_n$  по правилу:  $X_i=1$ , если элемент  $A_{ii}$  больше каждого из элементов  $i$ -й строки матрицы, иначе  $X_i=0$ . Найти также сумму элементов матрицы.

## **ЗАДАНИЕ 3**

### **Описание**

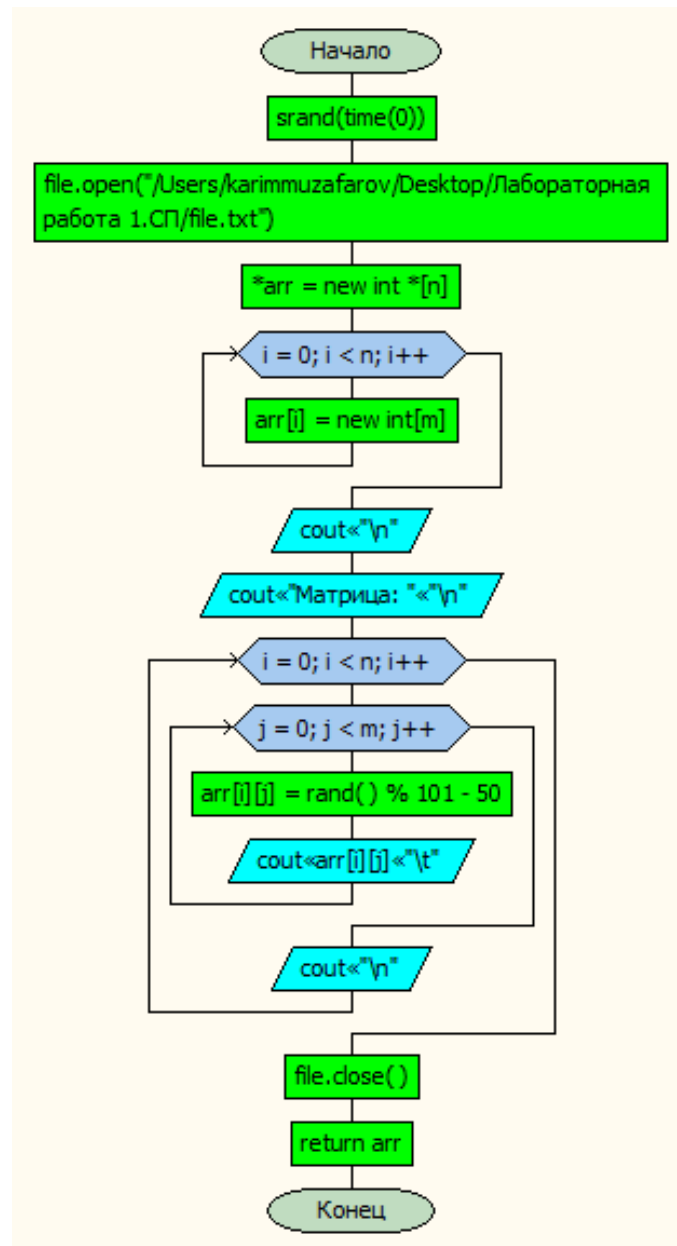
Изменить матрицу, заменив каждый отрицательный элемент, лежащий выше главной диагонали, его абсолютной величиной. Найти также сумму элементов главной диагонали.

### Использованные переменные

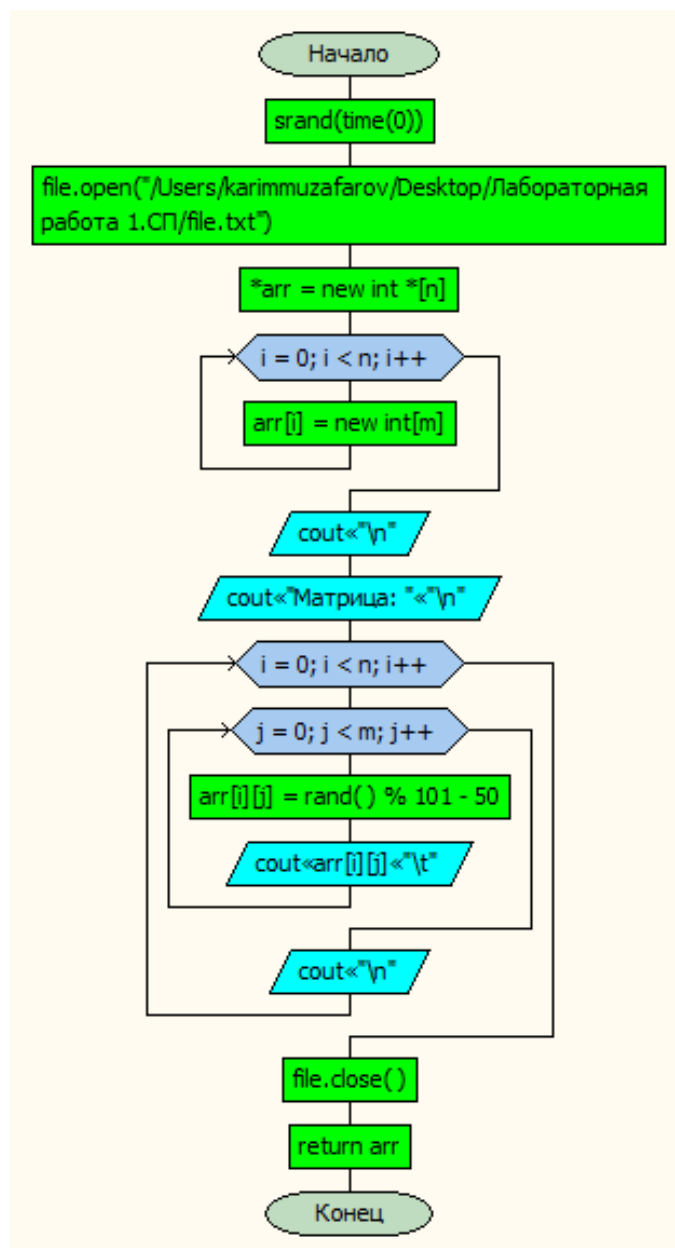
Имя переменной	Класс	Тип	Смысл
n	Глобальная	Целочисленный (int)	Количество строк массива
m	Глобальная	Целочисленный (int)	Количество столбцов массива
arr	Локальная	Целочисленный (int)	Массив создаваемый для первого задания
i	Локальная	Целочисленный (int)	Итерации цикла
j	Локальная	Целочисленный (int)	Итерации цикла
sum	Локальная	Целочисленный (int)	Сумма значений строк
counter	Локальная	Целочисленный (int)	Количество элементов
average	Локальная	Целочисленный (int)	Среднее арифметическое
k	Локальная	Целочисленный (int)	Количество неотрицательных элементов в строке
Arr	Локальная	Указатель первого порядка (int*)	Одномерный динамический массив
Arr1	Локальная	Указатель второго порядка (int**)	Двумерный динамический массив создаваемый для второго задания
Arr2	Локальная	Указатель второго порядка (int**)	Двумерный динамический массив создаваемый для третьего задания
count	Локальная	Целочисленный (int)	Счетчик для проверки условия в задании 2
sumMain	Локальная	Целочисленный (int)	Сумма элементов главной матрицы

## Блок-схема

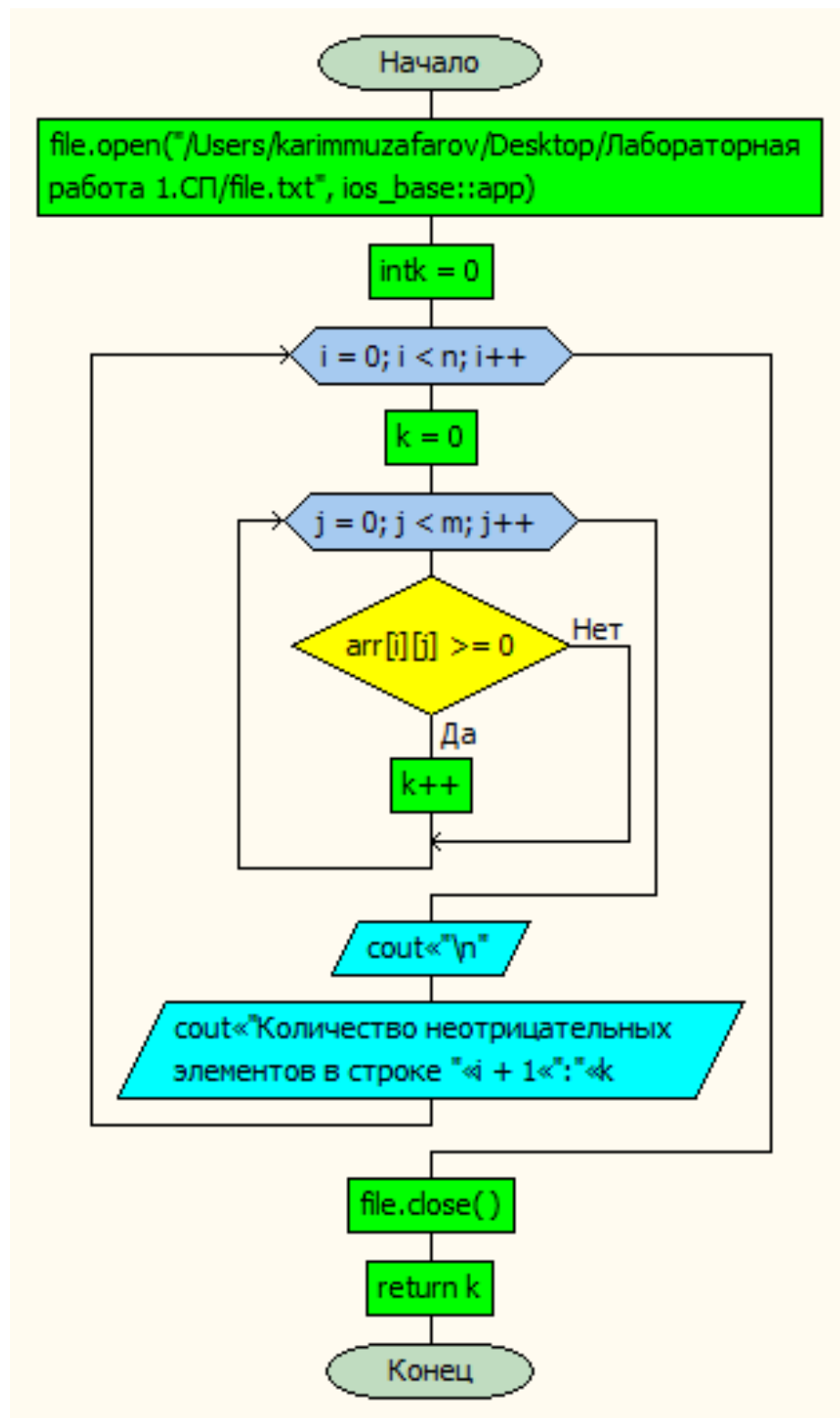
## CreateArray



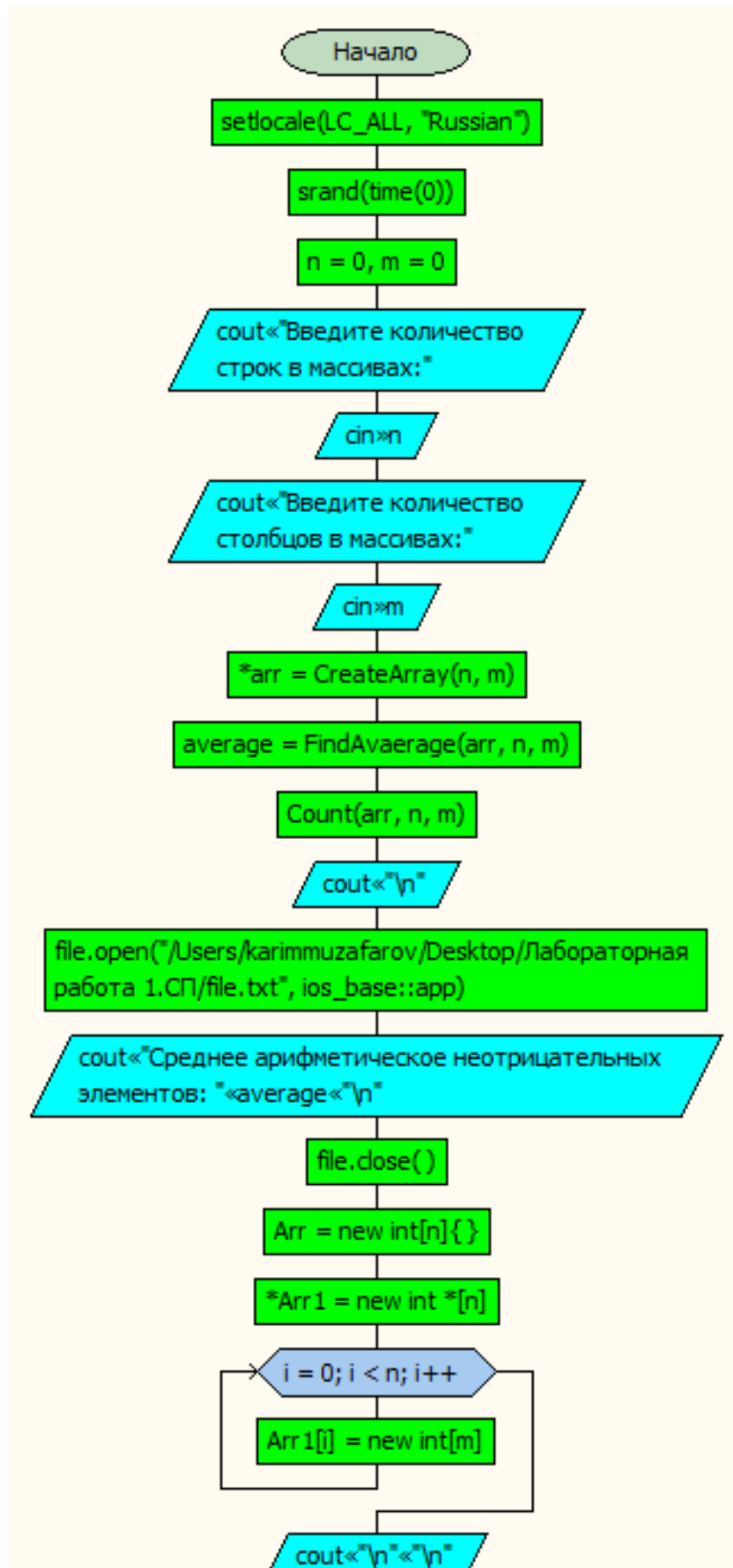
## FindAverage

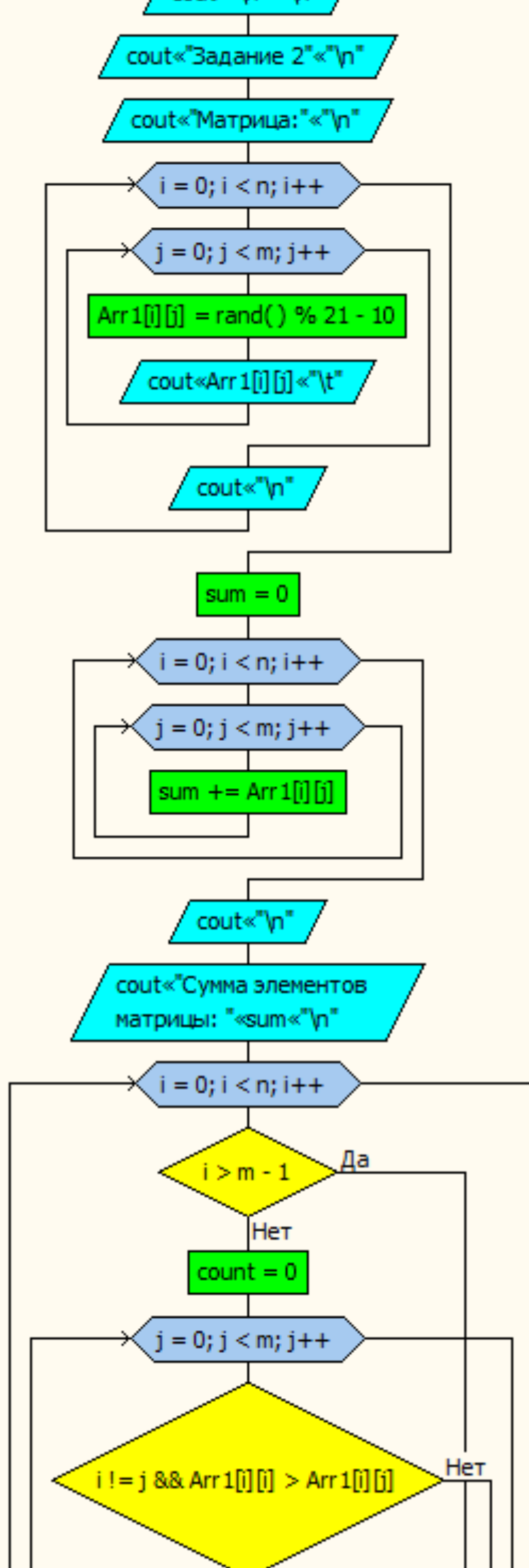


## Count

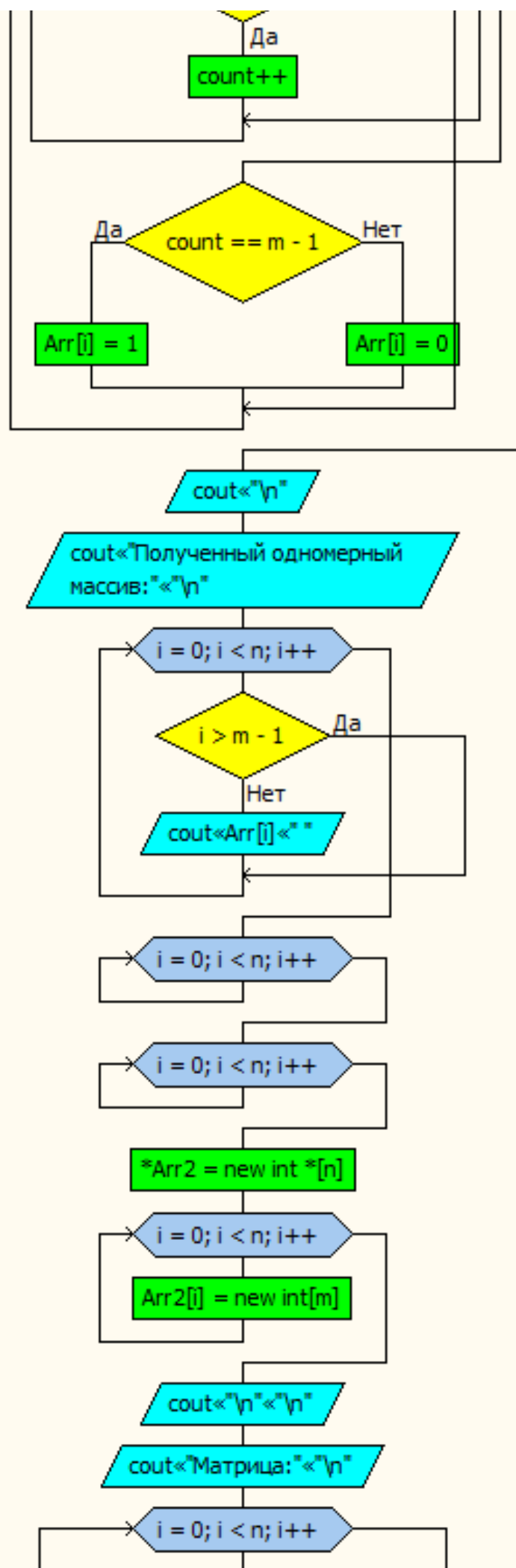


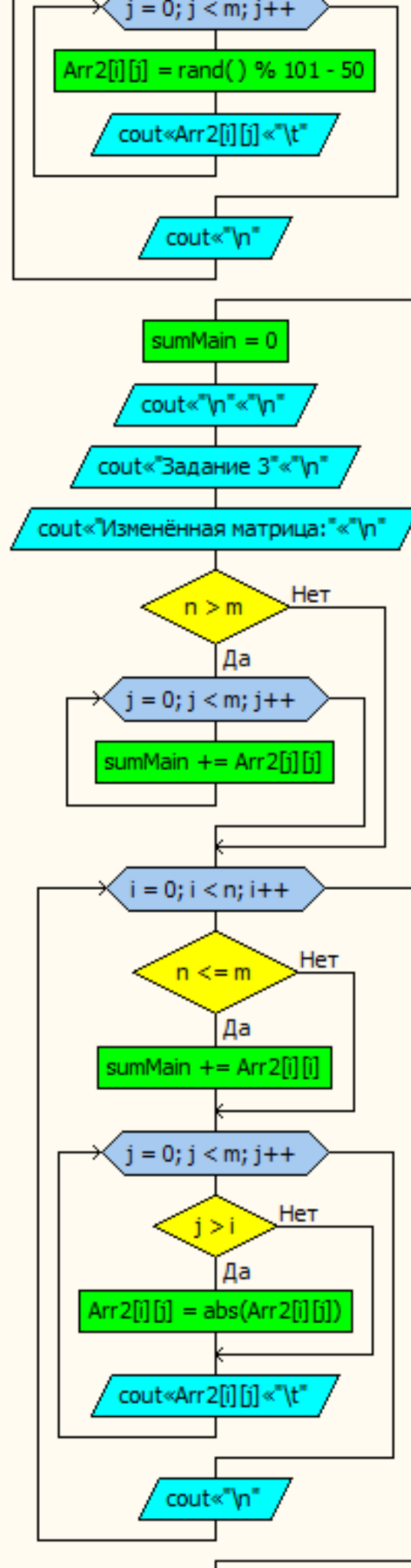
## Main

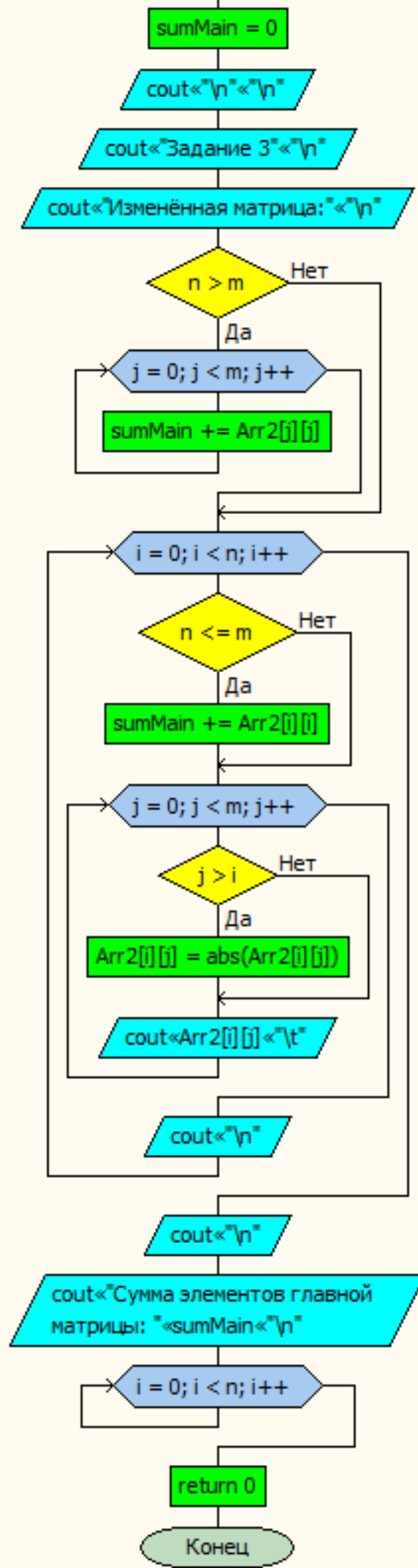












## Программа

```
#include <iostream>

#include<ctime>

#include<cstdlib>

#include<cmath>

#include<fstream>


using namespace std;

int** CreateArray(int n, int m)
{
    srand(time(0));

    ofstream file;

    file.open("/Users/karimmuzafarov/Desktop/Лабораторная работа 1.СП/file.txt");

    int** arr= new int* [n];

    for (int i = 0; i < n; i++)
    {
        arr[i] = new int[m];
    }


    cout << "\n";

    cout <<"Матрица: " << "\n";

    file << "\n";

    file <<"Матрица: " << "\n";

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

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

        {

            arr[i][j] = rand() % 101 - 50;
```

```

        cout << arr[i][j] << "\t";

        file << arr[i][j] << "\t";

    }

    cout << "\n";

    file << "\n";

}

file.close();

    return arr;

}

```

```

double FindAvaerage(int** arr, int n, int m)

```

```

{

    float sum = 0;

    float counter = 0;

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

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

        {

            if (arr[i][j] >= 0)

            {

                sum += arr[i][j];

                counter++;

            }

        }

    }

    float average = sum / counter;

    return average;

}

```

```

int Count(int** arr, int n, int m)
{
    ofstream file;

    file.open("/Users/karimmuzafarov/Desktop/Лабораторная работа 1.СП/file.txt", ios_base::app);

    int k=0;

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

        for (int j = 0; j < m; j++)
        {
            if (arr[i][j] >= 0)
                k++;
        }

        cout << "\n";

        cout<<"Количество неотрицательных элементов в строке " << i + 1 << ":" << k;

        file << "\n";

        file << "Количество неотрицательных элементов в строке " << i + 1 << ":" << k;
    }

    file.close();

    return k;
}

int main()
{
    setlocale(LC_ALL, "Russian");

    srand(time(0));

    int n = 0, m = 0;

```

```

cout << "Введите количество строк в массивах:";

cin >> n;

cout << "Введите количество столбцов в массивах:";

cin >> m;


int** arr = CreateArray(n, m);

float average = FindAverage(arr, n, m);

Count(arr, n, m);

cout << "\n";

ofstream file;

file.open("/Users/karimmuzafarov/Desktop/Лабораторная работа 1.СП/file.txt", ios_base::app);

cout << "Среднее арифметическое неотрицательных элементов: " << average << "\n";

file << "\n";

file << "Среднее арифметическое неотрицательных элементов: " << average << "\n";

file.close();

int* Arr = new int[n] {};

    int** Arr1 = new int* [n];

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

{

    Arr1[i] = new int[m];

}

cout << "\n" << "\n";

cout << "Задание 2" << "\n";

cout << "Матрица:" << "\n";

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

{

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

    {

        Arr1[i][j] = rand() % 21 - 10;

```

```

        cout << Arr1[i][j] << "\t";

    }

    cout << "\n";

}

int sum = 0;

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

{

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

    {

        sum += Arr1[i][j];

    }

}

cout << "\n";

cout << "Сумма элементов матрицы: " << sum << "\n";

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

{

    if (i > m - 1)

        continue;

    int count = 0;

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

    {

        if (i != j && Arr1[i][i] > Arr1[i][j])

            count++;

    }

    if (count == m - 1)

    {

        Arr[i] = 1;

```



```

    }

    else

    {

        Arr[i] = 0;

    }

}

cout << "\n";

cout << "Полученный одномерный массив:" << "\n";

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

{

    if (i > m - 1)

        continue;

    cout << Arr[i] << " ";

}

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

{

    delete[] Arr1[i];

}

delete[] Arr1;

delete[] Arr;

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

{

    delete[] arr[i];

}

delete[] arr;

int** Arr2 = new int* [n];

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

{

    Arr2[i] = new int[m];

```

```

}

cout << "\n" << "\n";

cout << "Матрица:" << "\n";

for (int i = 0; i < n; i++)
{
    for (int j = 0; j < m; j++)
    {
        Arr2[i][j] = rand() % 101 - 50;

        cout << Arr2[i][j] << "\t";
    }

    cout << "\n";
}

int sumMain = 0;

cout << "\n" << "\n";

cout << "Задание 3" << "\n";

cout << "Изменённая матрица:" << "\n";

if(n>m){
    for (int j = 0; j < m; j++)
        sumMain += Arr2[j][j];
}

for (int i = 0; i < n; i++)
{
    if(n<=m){
        sumMain += Arr2[i][i];
    }

    for (int j = 0; j < m; j++)
    {
        if (j > i)

            Arr2[i][j] = abs(Arr2[i][j]);
    }
}

```

```
        cout << Arr2[i][j] << "\t";
    }
    cout << "\n";
}
cout << "\n";
cout << "Сумма элементов главной матрицы: " << sumMain << "\n";
for (int i = 0; i < n; i++)
{
    delete[] Arr2[i];
}
delete[] Arr2;
return 0;
}
```

## Примеры работы программы

При  $n = 5$ ,  $m = 4$

```
Введите количество строк в массивах:5
Введите количество столбцов в массивах:4

Матрица:
50 -36 37 35
-7 -47 21 27
-33 -8 -8 47
-21 44 40 18
-4 43 -1 -21

Количество неотрицательных элементов в строке 1:3
Количество неотрицательных элементов в строке 2:2
Количество неотрицательных элементов в строке 3:1
Количество неотрицательных элементов в строке 4:3
Количество неотрицательных элементов в строке 5:1
Среднее арифметическое неотрицательных элементов: 36.2

Задание 2
Матрица:
-2 -9 0 3
-2 3 1 4
7 6 -3 -2
-1 -4 4 6
-8 -2 8 -10

Сумма элементов матрицы: -1

Полученный одномерный массив:
0 0 0 1

Матрица:
16 -49 28 49
10 -34 -48 45
11 50 47 15
7 -33 -30 -13
-21 4 -28 -31

Задание 3
Изменённая матрица:
16 49 28 49
10 -34 48 45
11 50 47 15
7 -33 -30 -13
-21 4 -28 -31

Сумма элементов главной матрицы: 16
Program ended with exit code: 0
```

```

file.txt — Изменено

Матрица:
50      -36      37      35
-7       -47      21      27
-33      -8       -8      47
-21      44       40      18
-4       43       -1     -21

Количество неотрицательных элементов в строке 1:3
Количество неотрицательных элементов в строке 2:2
Количество неотрицательных элементов в строке 3:1
Количество неотрицательных элементов в строке 4:3
Количество неотрицательных элементов в строке 5:1
Среднее арифметическое неотрицательных элементов: 36.2|

```

При  $n = 2$ ,  $m = 3$

```

Матрица:
-44 40 2
27 11 34

Количество неотрицательных элементов в строке 1:2
Количество неотрицательных элементов в строке 2:3
Среднее арифметическое неотрицательных элементов: 22.8

Задание 2
Матрица:
8 6 -10
-2 6 1

Сумма элементов матрицы: 9

Полученный одномерный массив:
1 1

Матрица:
35 45 16
-38 12 -23

Задание 3
Изменённая матрица:
35 45 16
-38 12 23

Сумма элементов главной матрицы: 47
Program ended with exit code: 0|

```

```
file.txt — Изменено

Матрица:
-44    40    2
 27    11   34

Количество неотрицательных элементов в строке 1:2
Количество неотрицательных элементов в строке 2:3
Среднее арифметическое неотрицательных элементов: 22.8
```

## Вывод

В этой лабораторной работе была написана программа, которая находит среднее арифметическое неотрицательных элементов матрицы, а также подсчитывает, сколько таких элементов в каждой отдельно взятой строке матрицы, получает массив  $X_1, X_2, \dots, X_n$  по правилу:  $X_i=1$ , если элемент  $A_{ii}$  больше каждого из элементов  $i$ -й строки матрицы, иначе  $X_i=0$ . Находит также сумму элементов матрицы, изменяет матрицу, заменив каждый отрицательный элемент, лежащий выше главной диагонали, его абсолютной величиной. Находит также сумму элементов главной диагонали. Для решения данной задачи были изучены основы работы с двумерными массивами, а также особенности функционирования циклов.