



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение  
высшего образования

«МИРЭА – Российский технологический университет»

**РТУ МИРЭА**

Институт информационных технологий

Кафедра вычислительной техники

## **ОТЧЕТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ №3**

**по дисциплине**

**«Алгоритмические основы обработки данных»**

Выполнил студент группы ИВБО-01-20

Д.А. Манохин

Принял старший преподаватель

Ю.С. Асадова

Практические работы выполнены

«\_\_»\_\_\_\_\_2021г.

«Зачтено»

«\_\_»\_\_\_\_\_2021г.

Москва 2021



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение  
высшего образования

«МИРЭА – Российский технологический университет»

**РТУ МИРЭА**

Институт информационных технологий

Кафедра вычислительной техники

Выполнено \_\_\_\_\_ /Д.А. Манохин/

Зачтено \_\_\_\_\_ /Ю.С. Асадова/

### **Задание на практическую работу №3**

Дисциплина: «Алгоритмические основы обработки данных»

Студент Манохин Дмитрий Александрович Шифр 20И2132 Группа ИББО-01-20

**1. Тема:** «Двумерные массивы».

**2. Срок сдачи студентом законченной работы:** 16.09.2021г.

**3. Исходные данные:**

Дана целочисленная матрица из  $n$  строк и  $m$  столбцов ( $1 < n \leq 100$ ,  $1 < m \leq 50$ ).

**4. Задание:**

Добавление строк. После каждой нулевой строки добавить массив  $b(m)$ .

**5. Содержание отчета:**

- титульный лист;
- задание;
- оглавление;
- введение;
- основные разделы отчета;
- заключение;
- список использованных источников;

Руководитель работы

Ю.С. Асадова

\_\_\_\_\_ «\_\_\_» \_\_\_\_\_ 2021г.  
подпись

Задание принял к исполнению

Д.А. Манохин

\_\_\_\_\_ «\_\_\_» \_\_\_\_\_ 2021г.  
подпись

# ОГЛАВЛЕНИЕ

Введение.....	4
Основной раздел.....	5
Заключение .....	9
Список использованных источников .....	10

## ВВЕДЕНИЕ

В данной практической работе требуется применить навыки работы с матрицами, а также с их индексами.

Постановка задачи:

Дана целочисленная матрица из  $n$  строк и  $m$  столбцов ( $1 < n \leq 100$ ,  $1 < m \leq 50$ ). Выполнить заданную операцию над матрицей. Добавление строк. После каждой нулевой строки добавить массив  $b(m)$ .

При этом должны выполняться следующие условия:

- Введены данные с клавиатуры (количество элементов матрицы и их значения).
- Порядок элементов матрицы не поменяется после работы алгоритма.
- Программа выводит результат своей работы: матрица с добавленным массивом  $b(m)$  после каждой нулевой строки.

## ОСНОВНОЙ РАЗДЕЛ

В начале программы объявляются три целочисленные переменные, а также динамический двумерный массив. Далее пользователь вводит значения количества строк и столбцов матрицы. После чего, с помощью цикла в цикле, происходит ввод значения каждого элемента двумерного массива. Также по нашему условию необходимо ввести элементы дополнительного массива, что и происходит в дополнительном цикле.

После обработки всех введенных данных начинается алгоритм нахождения нулевой строки в нашем двумерном массиве, происходит это с помощью вложенного цикла (цикл в цикле). В внешнем цикле, который используется для прохождения каждой строки матрицы, каждый проход присваивается некой переменной  $k$  значение равно единице. Далее в внутреннем цикле идет проверка: если элемент  $arr[i][j]$  не равен нулю, то переменной  $k$  присваивается значение ноль, это нужно для определения полная ли строка состоит из нулей.

Как только внутренний цикл закончил перебор значений, то идет проверка на значение переменной  $k$ . Если оно равно единице, то начинается еще один внутренний цикл с перебором значений от количества строк до значения переменной, отвечающей за текущий ход внешнего цикла. Присваиваем каждой строке от конца, значение предыдущей строке матрицы. Таким образом мы смещаем все строки снизу на одну вниз, до нашей строки, состоящей из нулей.

После вставляем массив  $b[m]$  в строку, следующую после одних нулей. И не забываем увеличить значение, отвечающее, за количество строк динамического двумерного массива на единицу.

Если алгоритм отработал, то осталось только вывести матрицу. Выводим строку: “Полученный массив: “ и с помощью вложенного цикла с новой строки выводим значение каждого элемента с расстоянием в один пробел между каждым элементом.

Блок – схема алгоритма представлена на [рисунке 1](#).

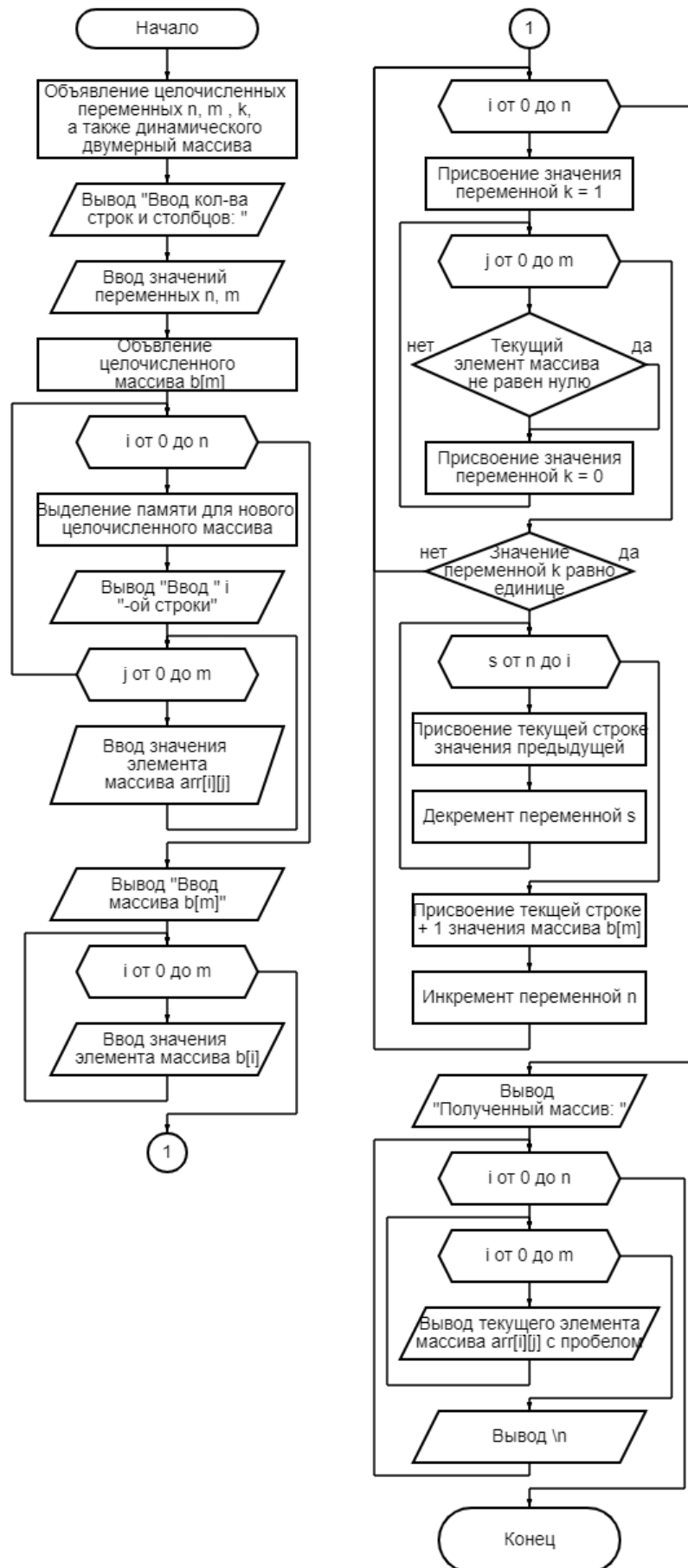


Рисунок 1 – Блок – схема алгоритма

Исходный код программы представлен в Листинге А.1.

*Листинг А.1 – Основной код*

```
#include <iostream>
using namespace std;

int main()
{
    int n, m, k, **arr;
    cout << "Ввод кол-ва строк и столбцов: ";
    cin >> n >> m;
    arr = new int* [n];
    int *b = new int [m];
    for(int i = 0; i < n; i++){
        arr[i] = new int [m];
        cout << "Ввод " << i << "-ой строки массива: ";
        for(int j = 0; j < m; j++){
            cin >> arr[i][j];
        }
    }
    cout << "Ввод массива b[m]: ";
    for(int i = 0; i < m; i++){
        cin >> b[i];
    }
    for(int i = 0; i < n; i++){
        k = 1;
        for(int j = 0; j < m; j++){
            if(arr[i][j] != 0){
                k = 0;
            }
        }
        if(k == 1){
            for(int s = n; s > i; s--){
                arr[s] = arr[s-1];
            }
            arr[i+1] = b;
            n++;
        }
    }
    cout << "Полученный массив:\n";
    for(int i = 0; i < n; i++){
        for(int j = 0; j < m; j++){
            cout << arr[i][j] << " ";
        }
        cout << endl;
    }
    return 0;
}
```

Пример работы программы представлен на рисунке [2](#) и рисунке [3](#).

```
Ввод кол-ва строк и столбцов: 3 4
Ввод 0-ой строки массива: 3 2 1 4
Ввод 1-ой строки массива: 0 0 0 0
Ввод 2-ой строки массива: 5 6 7 1
Ввод массива b[m]: 1 2 3 4
Полученный массив:
3 2 1 4
0 0 0 0
1 2 3 4
5 6 7 1
```

Рисунок 2 – Пример работы с нулевой строки

```
Ввод кол-ва строк и столбцов: 3 5
Ввод 0-ой строки массива: 3 4 6 0 9
Ввод 1-ой строки массива: 1 3 4 5 7
Ввод 2-ой строки массива: 7 5 3 2 2
Ввод массива b[m]: 1 1 1 1 1
Полученный массив:
3 4 6 0 9
1 3 4 5 7
7 5 3 2 2
```

Рисунок 3 – Пример работы без нулевой строкой



## **ЗАКЛЮЧЕНИЕ**

В результате выполнения данной практической работы были закреплены основные знания о составлении алгоритмов, использующие вложенные циклы, а также о работе с одномерными и двумерными динамическими массивами. Были закреплены навыки использования основных библиотек языка программирования C++.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Кубенский А.А. Структуры и алгоритмы обработки данных: объектноориентированный подход и реализация на С++ / А.А. Кубенский.— М.: БХВ-Петербург, 2017.— 300 с.
2. Стивен Прата. Язык программирования С++ (С++11). Лекции и упражнения, 6-е издание — М.: Вильямс, 2012. — 1248 с.
3. Седжвик, Р. Алгоритмы на С++ / Р. Седжвик.— М.:Вильямс, 2017.— 1056 с.