Бюджетное учреждение высшего образования

Ханты-Мансийского автономного округа

«Сургутский государственный университет»

Политехнический институт

Кафедра автоматики и компьютерных систем

**Отчет**

по лабораторной работе № 5 «Многомерные массивы»

по дисциплине «Программирование и основы алгоритмизации»

Выполнил: Скорба Р.В.

студент группы 609-11

Проверил: Гришмановский П. В.

доцент кафедры автоматики и

компьютерных систем

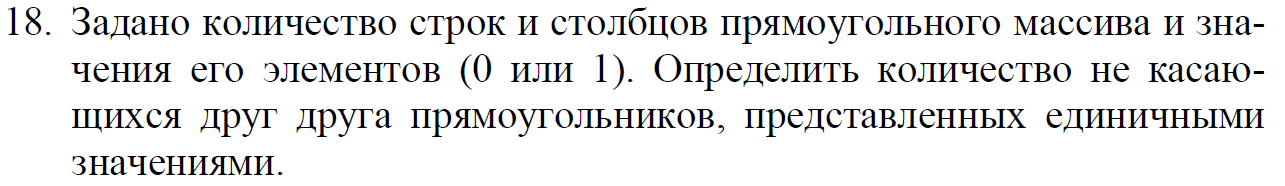
Сургут

2022 г.

**Цель работы:**

Целью работы является закрепление теоретических знаний и приобретение практических навыков использования многомерных массивов.

**Задание:**

 Решить задачу в соответствии с индивидуальным заданием, выданным преподавателем, используя многомерные массивы.

**Вариант №18:**

**Вариант 18.**

**Формальное описание задачи**

В начале пользователь вводит размер массива, по условию он не сказан, а значит не более 15 на 15.

После чего пользователь полностью заполняет массив 0 или 1. В программе так же присутствует проверка ввода. В случае если пользователь введет числа отличные от 0 или 1, то программа выдаст ошибку и попросит ввести значение заново.

Суть программы заключается в том сосчитать в массиве кол-во прямоугольников, прямоугольники не должны касаться друг друга. Прямоугольником считается единица, прямоугольник может состоять из нескольких единиц стоящих в ширину или высоту, а так же образуя один большой прямоугольник, если внутри по ширине и высоте отсутствуют нули. В программе не используются какие-либо математические формулы, кроме формулы нахождения площади прямоугольника. В остальном там применяется сравнение каждого элемента массива.

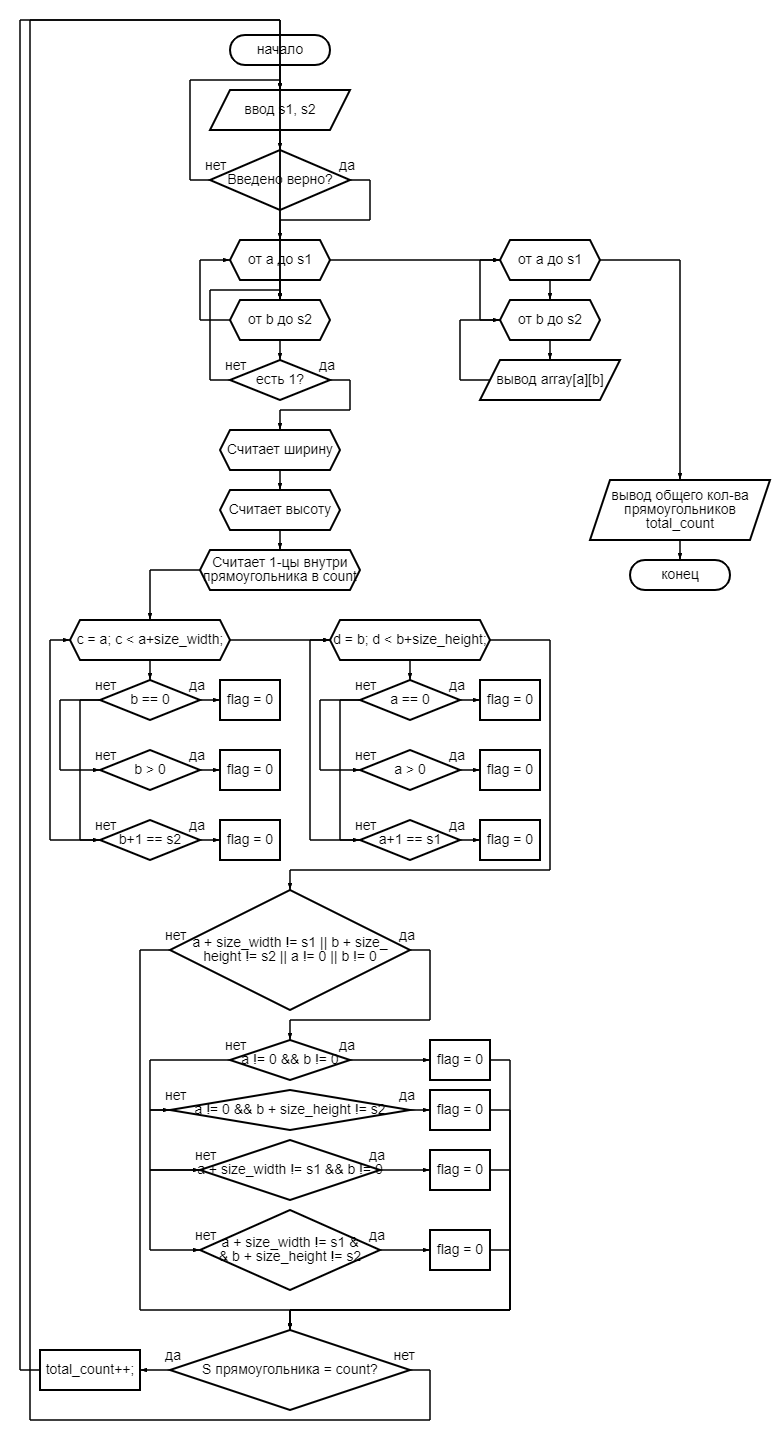
Программа проверяет поочередно каждый элемент массива на наличие единицы, если единица найдена, то она проверяет чтобы сверху, и слева не было единиц, а так же по диагонали слева вверху, т.к. эти элементы мы уже проверили. Это избавляет программу от большого кол-ва повторяющихся итераций.  
Если все условия истинны, то программа начинает считать кол-во единиц в ширину, а так же в высоту.  
Далее программа должна проверить, чтобы внутри получившегося прямоугольника не было нулей, проверка происходит путем счета кол-ва единиц внутри прямоугольника и сравнивается с тем кол-во которое должно получится, это значение получается по формуле площади прямоугольник, ширину умножив на высоту:

Далее чтобы прямоугольник можно было засчитать в ответ необходимо проверить, чтобы вокруг он не пересекался с другими прямоугольниками.   
Для этого запускается проверка на наполненность нулями по горизонтали и вертикали. Она проверяет поочередно каждый элемент массива вокруг прямоугольника на то, чтобы вокруг были нули, а так же отдельная проверка на то заполнен ли массив нулями по диагонали. В случае если везде все условия выполнились, то идет сравнение счетчика Count и площади прямоугольника. Если они равны, то прямоугольник действительный и счетчик увеличивается на 1, а все остальные переменные обнуляются и цикл повторяется для следующего элемента массива.  
 После того как все элементы будут проверены, для визуализации программа выводит на экран матрицу двумерного массива, для того чтобы было легче проверить на правильность выведенное значение. И выводит в конце общее кол-во прямоугольников.

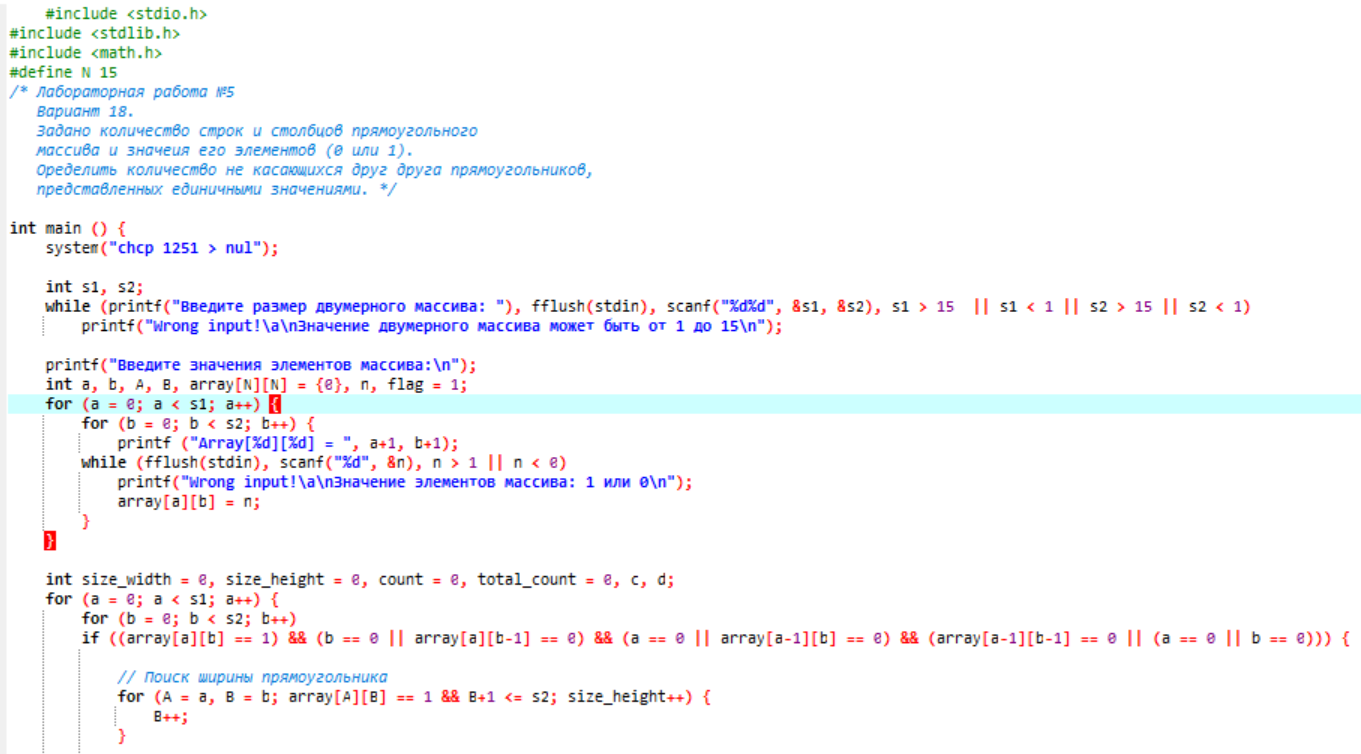
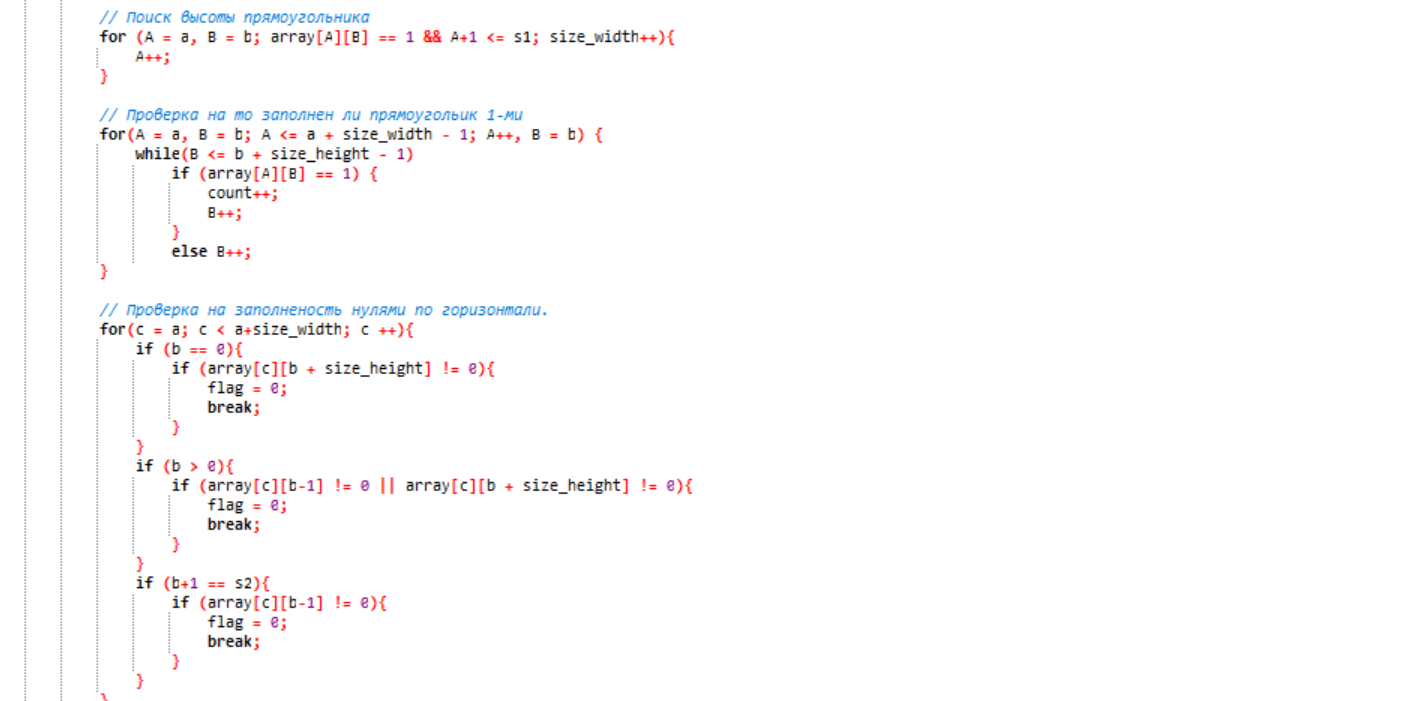
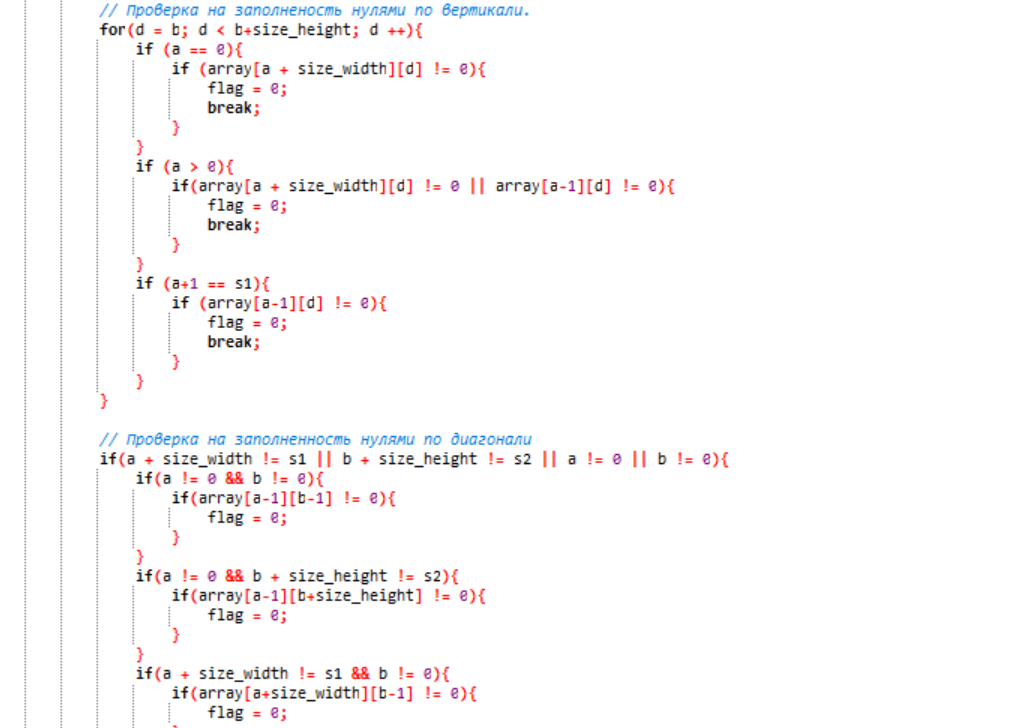
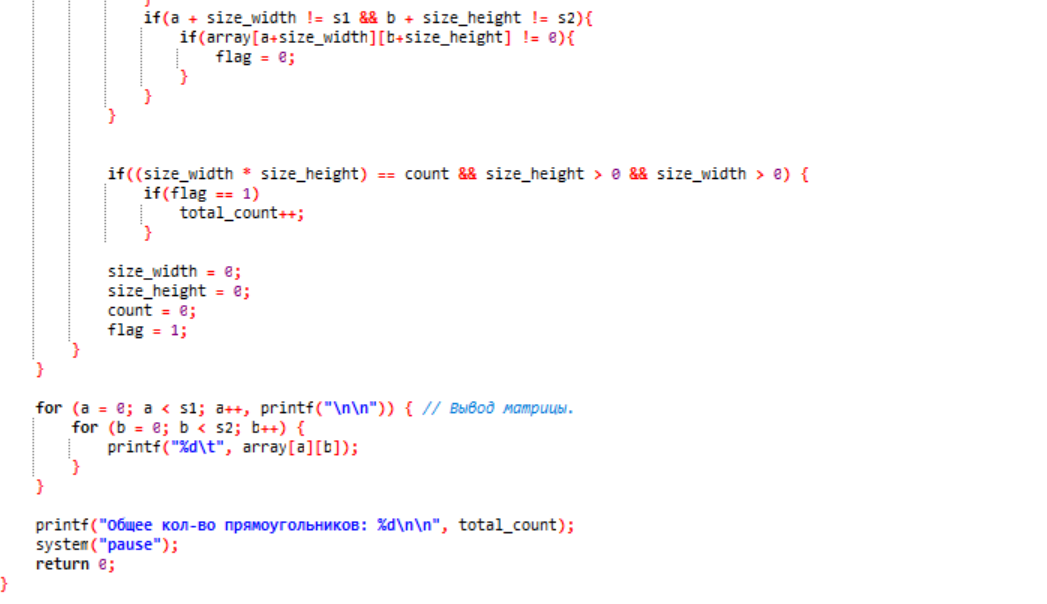
**Алгоритм программы.**

1. Ввести размер массива
2. Ввести элементы массива
3. Поиск единицы
4. Считает ширину прямоугольника
5. Считает высоту прямоугольник
6. Считает кол-во единиц внутри прямоугольника
7. Проверка на наполненность нулями по горизонтали
8. Проверка на наполненность нулями по вертикали
9. Проверка на наполненность нулями по диагонали
10. Считает площадь прямоугольника и сравнивает с пунктом 6. если верно, то счетчик увеличивается на 1
11. Проверка для каждого элемента массива пункты 3-10
12. Вывод матрицы массива
13. Вывод общего кол-во прямоугольников

**Блок-схема алгоритма**



**Листинг программы**

**Пояснение к программе.**

Программа состоит по большей степени из логических операторов if и циклов for, в данной задаче нет необходимости пользоваться сложными математическими формулами, все решение завязано на сравнении значений 1 или 0.

С помощью цикла While в программе осуществляется проверка чисел на адекватность.

Так же в конце программы для визуализации выводится матрица двумерного массива.  
В программе используется тип данных int, потому что все числа у нас целые 1 или 0, так же в операциях умножения и сложения используются целочисленные числа.

**Вывод:** Большое кол-во условий позволяет сократить большое кол-во повторяющихся итераций и ускорить работу программы.