

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра Вычислительной техники

ОТЧЕТ
по лабораторной работе № 3
по дисциплине «Программирование»
ТЕМА: «ОБРАБОТКА ДВУМЕРНЫХ МАССИВОВ»

Студент гр. 3311

Шарпинский Д. А.

Преподаватель

Хахаев И. А.

Санкт-Петербург

2023

Цель работы.

Целью работы является изучение двумерных массивов и получение практических навыков в их обработке.

Задание (вариант 14):

Заполнить двумерный массив размером $N \times N$ числами 1, 2, ..., N^2 , расположенными в нем по спирали ($N < 10$)

Постановка задачи и описание решения

Конечная задача: Заполнить двумерный массив размером $N \times N$ числами от 1 до N^2 так, чтобы числа были расположены в массиве в виде спирали. Каждое число должно быть уникальным.

Объяснение алгоритма для написания кода к решению этой задачи по шагам:

Объявляем и инициализируем переменные, которые будут использоваться в коде:

- `arr[10][10]`: Двумерный массив размером 10×10 для хранения чисел.
- `value`: Начальное значение для заполнения массива (1).
- `top`, `bottom`, `left`, `right`: Переменные для отслеживания границ массива.
- `col`, `row`: Индексы столбца и строки для текущей ячейки.
- `i`, `j`: Индексы для итерации по массиву.
- `n`: Переменная, в которую пользователь вводит размерность массива N .

1. Пользователь вводит значение n , которое определяет размерность массива ($N \times N$).
2. Инициализируем `value` (начальное значение для заполнения) и границы массива (`top`, `bottom`, `left`, `right`) для начала спирали.
3. Используем цикл `while`, который будет выполняться до тех пор, пока границы спирали не пересекутся. Этот цикл заполняет массив числами в спиральном порядке.
4. Внутри цикла используются четыре вложенных цикла `for`, каждый из которых заполняет одну из сторон спирали. Эти циклы выполняют следующее:
5. Первый цикл заполняет верхнюю сторону спирали от `left` до `right`.
6. Второй цикл заполняет правую сторону спирали от `top` до `bottom`.
7. Третий цикл заполняет нижнюю сторону спирали от `right` до `left`, только если `top` все еще меньше или равно `bottom`.
8. Четвертый цикл заполняет левую сторону спирали от `bottom` до `top`, только если `left` все еще меньше или равно `right`.
9. После заполнения одной стороны спирали соответствующие границы (`top`, `bottom`, `left`, `right`) обновляются, чтобы двигаться дальше внутрь спирали.
10. После выполнения цикла спирально заполняется весь массив числами.
11. Затем используется два вложенных цикла `for`, чтобы вывести массив на экран. Внешний цикл идет от 0 до $n-1$ (по строкам), и внутренний цикл идет от 0 до $n-1$ (по столбцам).
12. В каждой ячейке массива выводится число с использованием `printf` с форматированием `%4d`, чтобы числа были выровнены по ширине.
13. После вывода одной строки происходит перевод строки (`\n\n`), чтобы разделить строки в массиве.
14. Программа завершает выполнение.

Этот алгоритм позволяет заполнить двумерный массив числами от 1 до N^2 в виде спирали и вывести его на экран.

Описание переменных

№	Имя переменной	Тип	Назначение
1	arr	int[10][10]	Двумерный массив для хранения чисел.
2	value	int	Начальное значение для заполнения массива.
3	top	int	Верхняя граница текущей спирали.
4	bottom	int	Нижняя граница текущей спирали.
5	left	int	Левая граница текущей спирали.
6	right	int	Правая граница текущей спирали.
7	col	int	Индекс столбца текущей ячейки.
8	row	int	Индекс строки текущей ячейки.
9	i	int	Индекс для итерации по строкам массива.
10	j	int	Индекс для итерации по столбцам массива.
11	n	int	Размерность массива (NxN), введенная пользователем.

Контрольные примеры

Пример 1:

Входные данные: $N = 9$

Результат:

```
1  2  3  4  5  6  7  8  9
32 33 34 35 36 37 38 39 10
31 56 57 58 59 60 61 40 11
30 55 72 73 74 75 62 41 12
29 54 71 80 81 76 63 42 13
28 53 70 79 78 77 64 43 14
27 52 69 68 67 66 65 44 15
26 51 50 49 48 47 46 45 16
25 24 23 22 21 20 19 18 17
```

Пример 2:

Входные данные: $N = 6$

Результат:

1	2	3	4	5	6
20	21	22	23	24	7
19	32	33	34	25	8
18	31	36	35	26	9
17	30	29	28	27	10
16	15	14	13	12	11

Пример 3:

Входные данные: $N = 1$

Результат:

1

Пример 4:

Входные данные: $N = 0$

Результат:

Примеры выполнения программы

```
Enter the number of rows and cols: 9
```

```
Your array is:
```

```
  1  2  3  4  5  6  7  8  9
32 33 34 35 36 37 38 39 10
31 56 57 58 59 60 61 40 11
30 55 72 73 74 75 62 41 12
29 54 71 80 81 76 63 42 13
28 53 70 79 78 77 64 43 14
27 52 69 68 67 66 65 44 15
26 51 50 49 48 47 46 45 16
25 24 23 22 21 20 19 18 17
```

```
Process returned 0 (0x0)   execution time : 2.406 s
Press any key to continue.
```

```
Enter the number of rows and cols: 6
```

```
Your array is:
```

```
  1  2  3  4  5  6
20 21 22 23 24  7
19 32 33 34 25  8
18 31 36 35 26  9
17 30 29 28 27 10
16 15 14 13 12 11
```

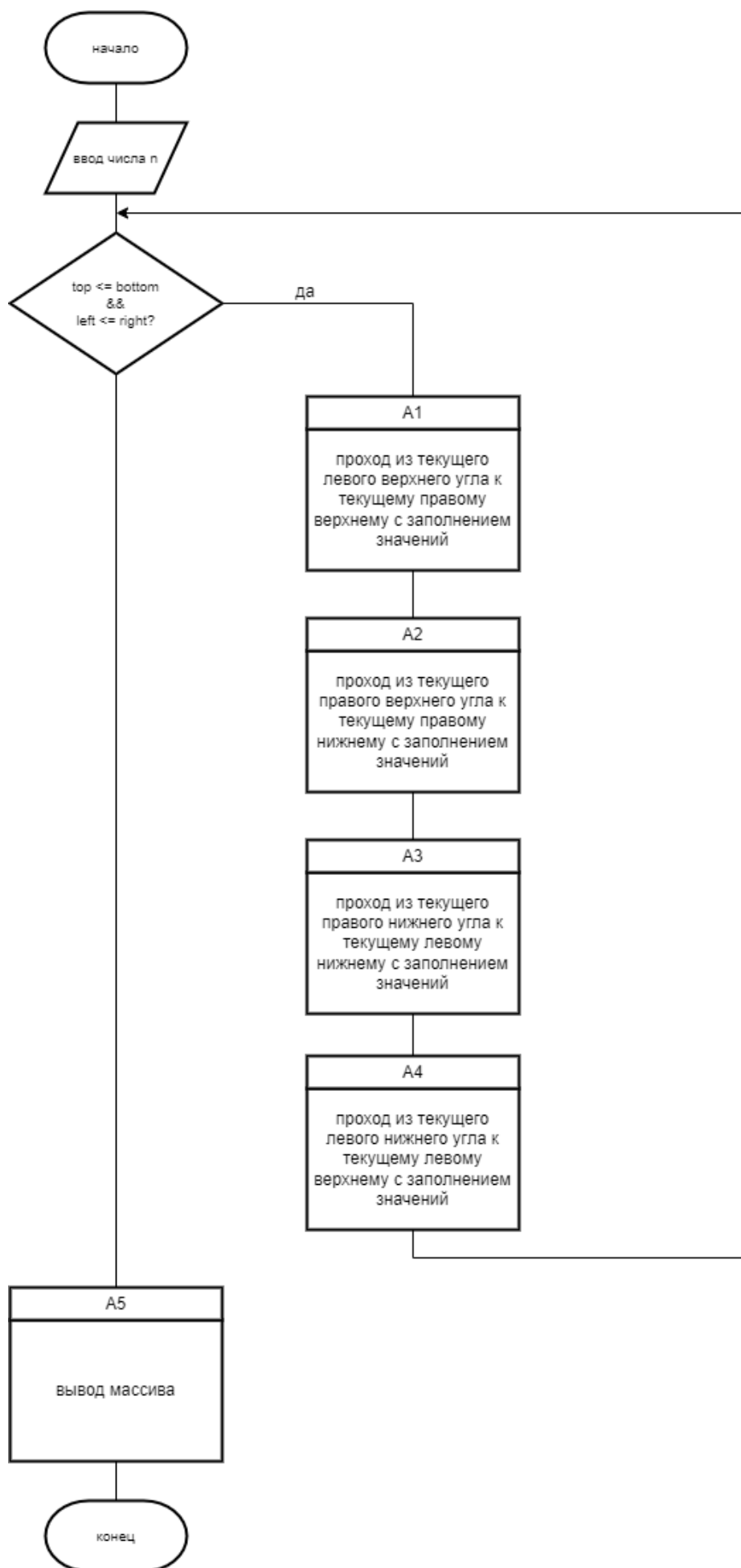
```
Process returned 0 (0x0)   execution time : 2.531 s
Press any key to continue.
```

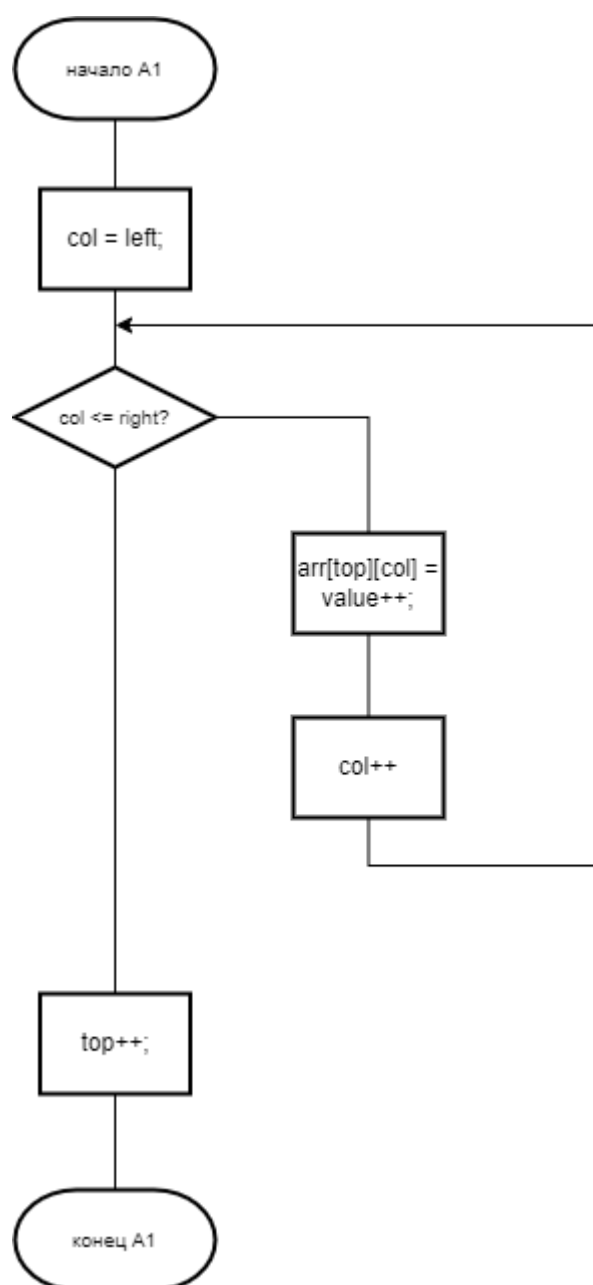
```
Enter the number of rows and cols: 1
Your array is:
1

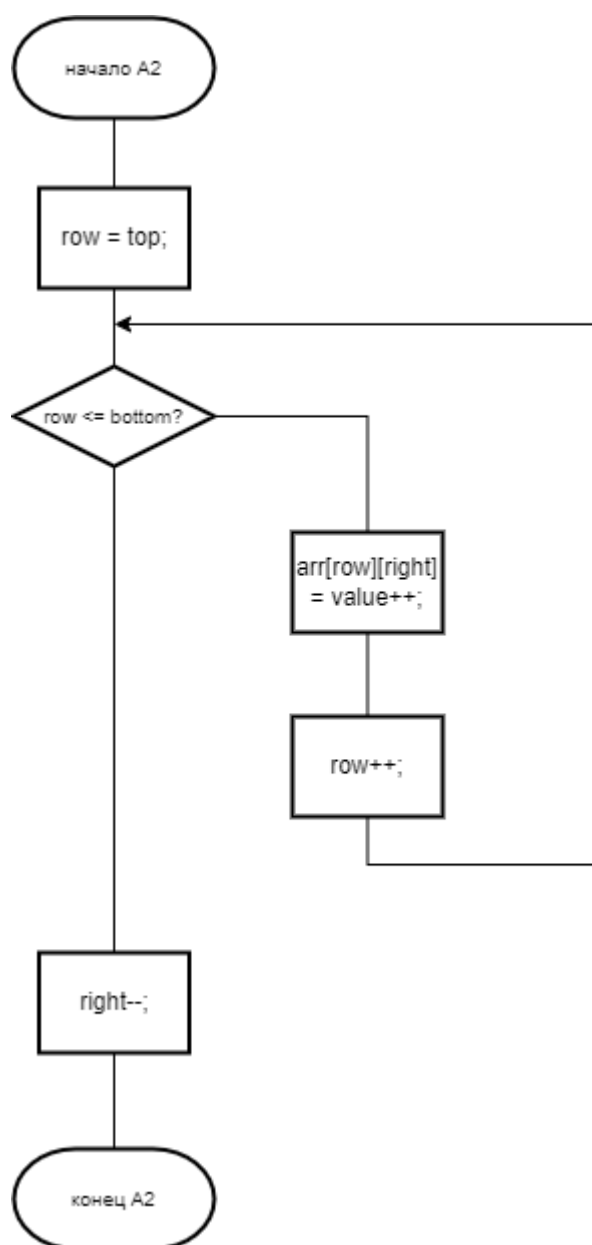
Process returned 0 (0x0)   execution time : 1.375 s
Press any key to continue.

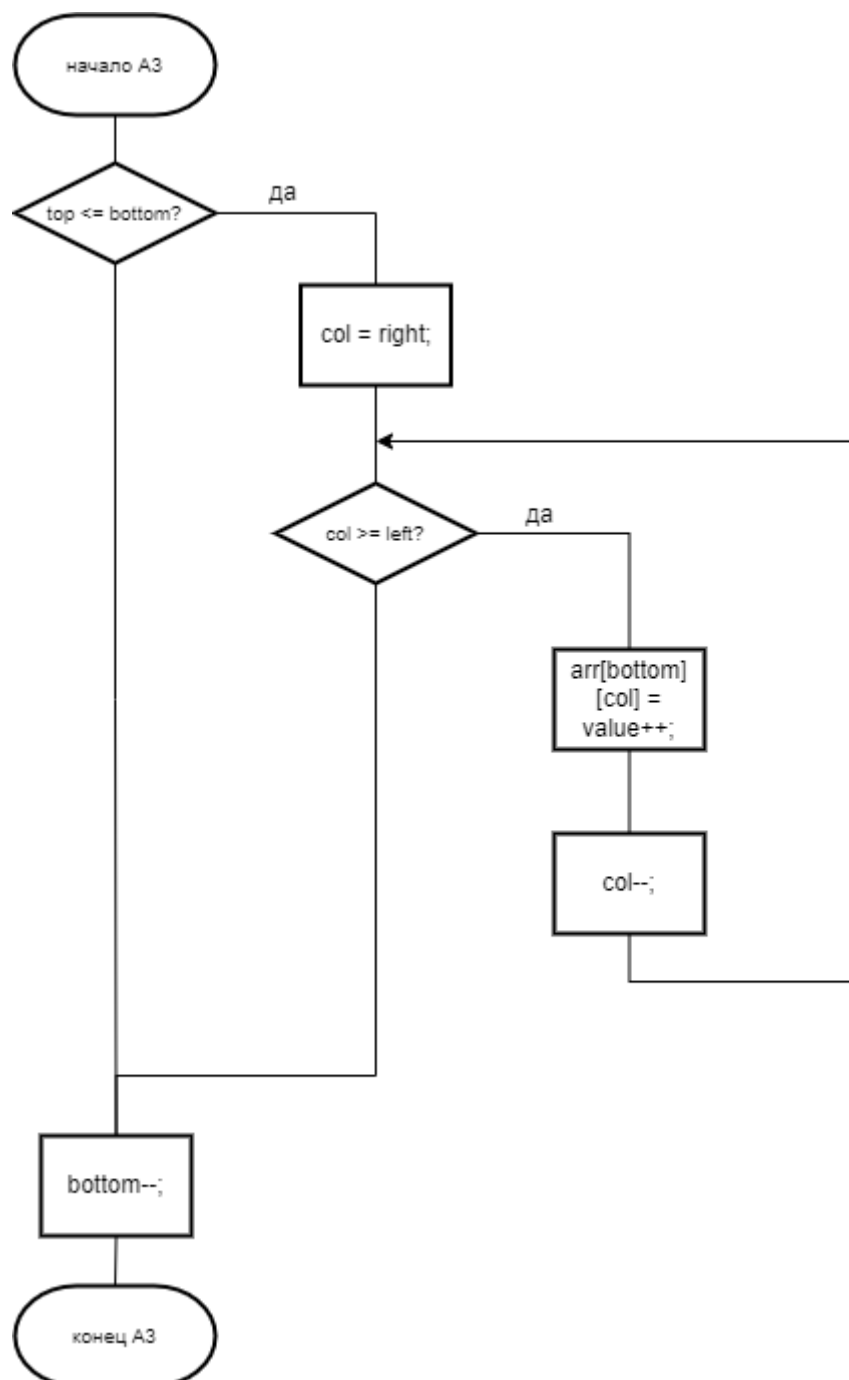
Enter the number of rows and cols: 0
Your array is:

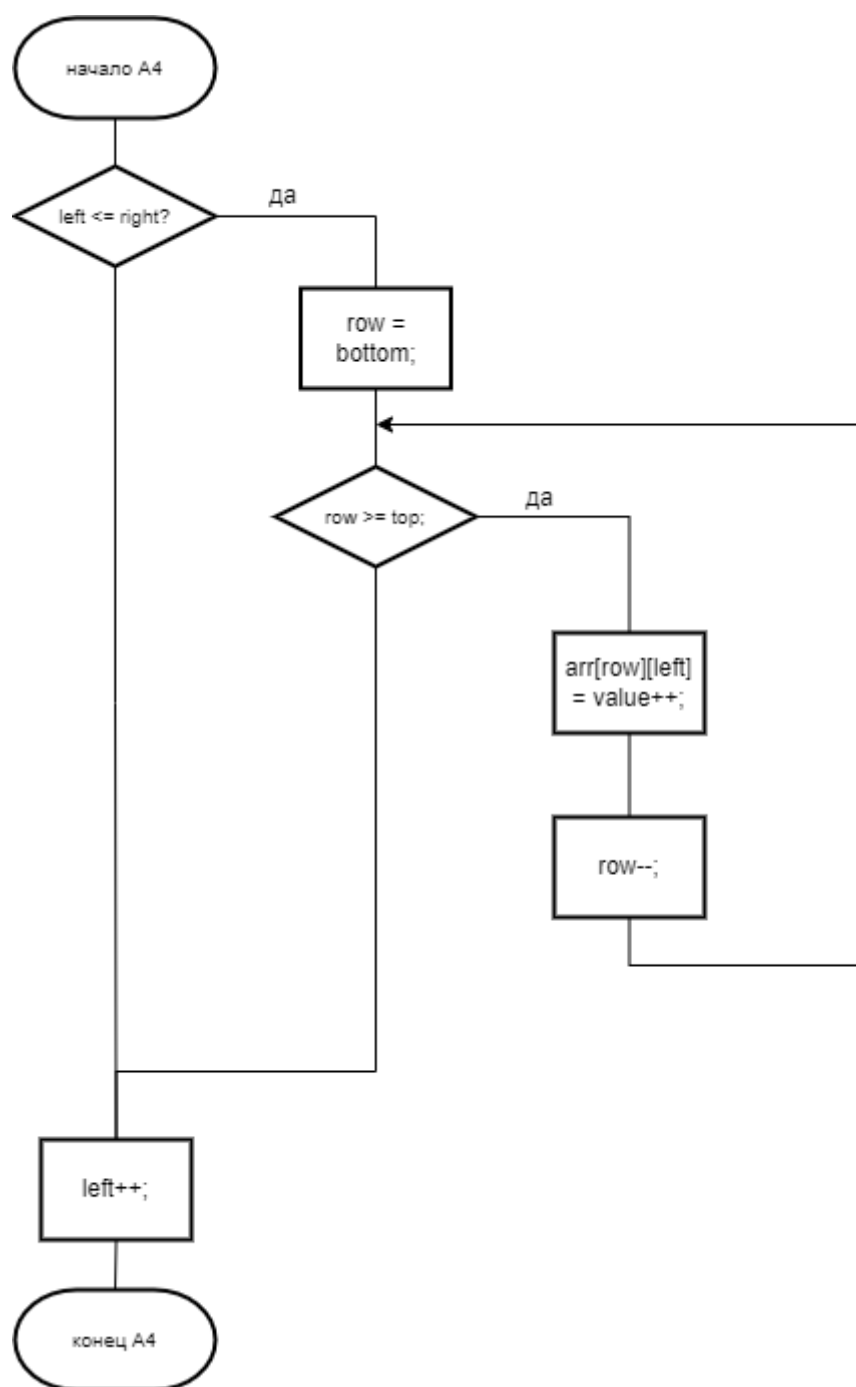
Process returned 0 (0x0)   execution time : 0.938 s
Press any key to continue.
```

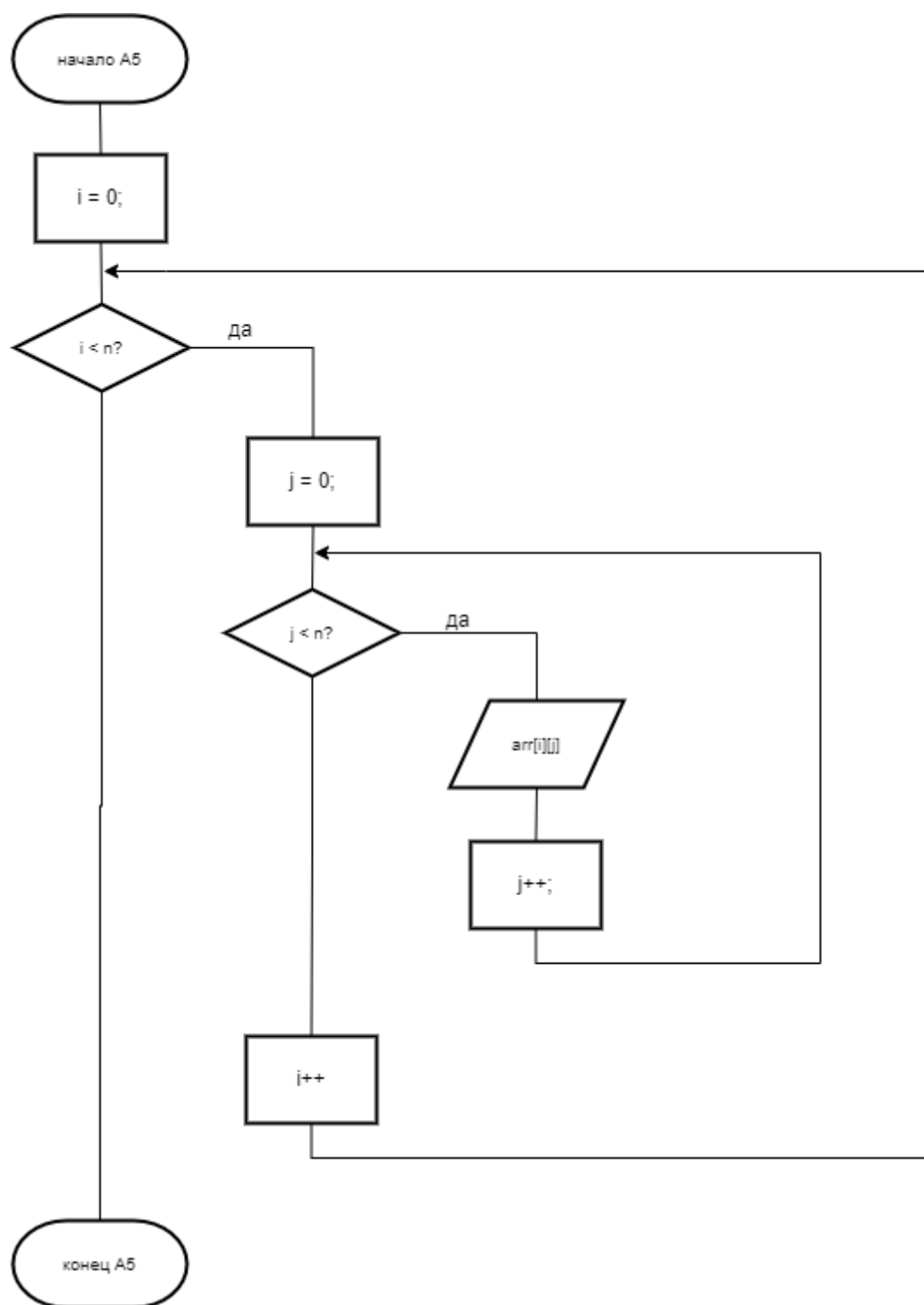












```

1  #include <stdio.h>
2
3  #define N 10
4
5  int main() {
6      int arr[N][N], value, top, bottom, left, right, col, row, i, j, n;
7
8      printf("Enter the size N of the square array (N < 10): ");
9      scanf("%d", &n);
10
11     if (n >= 10) {
12         printf("incorrect array size");
13         return 1;
14     }
15
16     printf("\nYour array is: \n\n");
17
18     value = 1;
19     top = 0;
20     bottom = n - 1;
21     left = 0;
22     right = n - 1;
23
24     while (top <= bottom && left <= right) {
25
26         for (col = left; col <= right; col++) {
27             arr[top][col] = value++;
28         }
29         top++;
30
31         for (row = top; row <= bottom; row++) {
32             arr[row][right] = value++;
33         }
34         right--;
35
36         if (top <= bottom) {
37             for (col = right; col >= left; col--) {
38                 arr[bottom][col] = value++;
39             }
40             bottom--;
41         }
42
43         if (left <= right) {
44             for (row = bottom; row >= top; row--) {
45                 arr[row][left] = value++;
46             }
47             left++;
48         }
49     }
50
51     for (i = 0; i < n; i++) {
52         for (j = 0; j < n; j++) {
53             printf("%4d", arr[i][j]);
54         }
55         printf("\n\n");
56     }
57
58     return 0;
59 }
60

```

Выводы:

В результате выполнения работы изучены двумерные массивы и получены практические навыки в их обработке.