

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

ОТЧЕТ
по лабораторной работе № 4
по дисциплине «Программирование»
Тема: Применение функций

Студент гр. 3311

Баймухамедов Р.Р.

Преподаватель

Хахаев И. А.

Санкт-Петербург

2023

Цель работы.

Целью лабораторной работы является изучение применения функций на языке программирования Си, их создание, вызов и использование для решения различных задач.

Задание (вариант 15)

Ввести построчно элементы двумерного массива чисел заданных размеров. Вывести исходный массив. Из строк исходного массива, в которых чередуются четные и нечетные числа, сформировать столбцы результирующего массива. Элементы в столбцах должны быть расположены в порядке обратном их расположению в строках. Вывести сформированный массив. Вывод массива и формирование столбца массива оформить в виде функций.

Постановка задачи и описание решения

Для выполнения данной лабораторной работы необходимо разработать программу, которая будет получать значение количества строк и столбцов исходного массива и его значения. Затем с помощью функции проверять условие (чередование четных и нечетных значений в строке) и формировать из строк исходного массива столбцы результирующего массива, записанные в обратном порядке, и выводить их.

1. Запросить у пользователя значение количества строк и столбцов в исходном массиве и сохранить эти значения в переменных `nr` и `nc`.
2. Запросить у пользователя значения столбцов и строк исходного массива.
3. (Функция) При начале проверки строки присваивать `flag` значение 1, после поочередно проверять элементы строки на условие чередования чётности и нечётности и при провале – изменять значение `flag` на 0.
4. (Функция) Если по окончании проверки строки значение `flag` осталось 1, то преобразуем эту строку в столбец результирующего массива
5. (Функция) Выводим результирующий массив.

Описание переменных

Функция void output (int ar[][size], int nr, int nc)

№	Имя переменной	Тип	Назначение
1	column	int	Переменная в цикле, отвечающая за столбец массива.
2	row	int	Переменная в цикле, отвечающая за строку массива.

int make_column (int ar[][size],int res[size][size],int nr,int nc,int j)

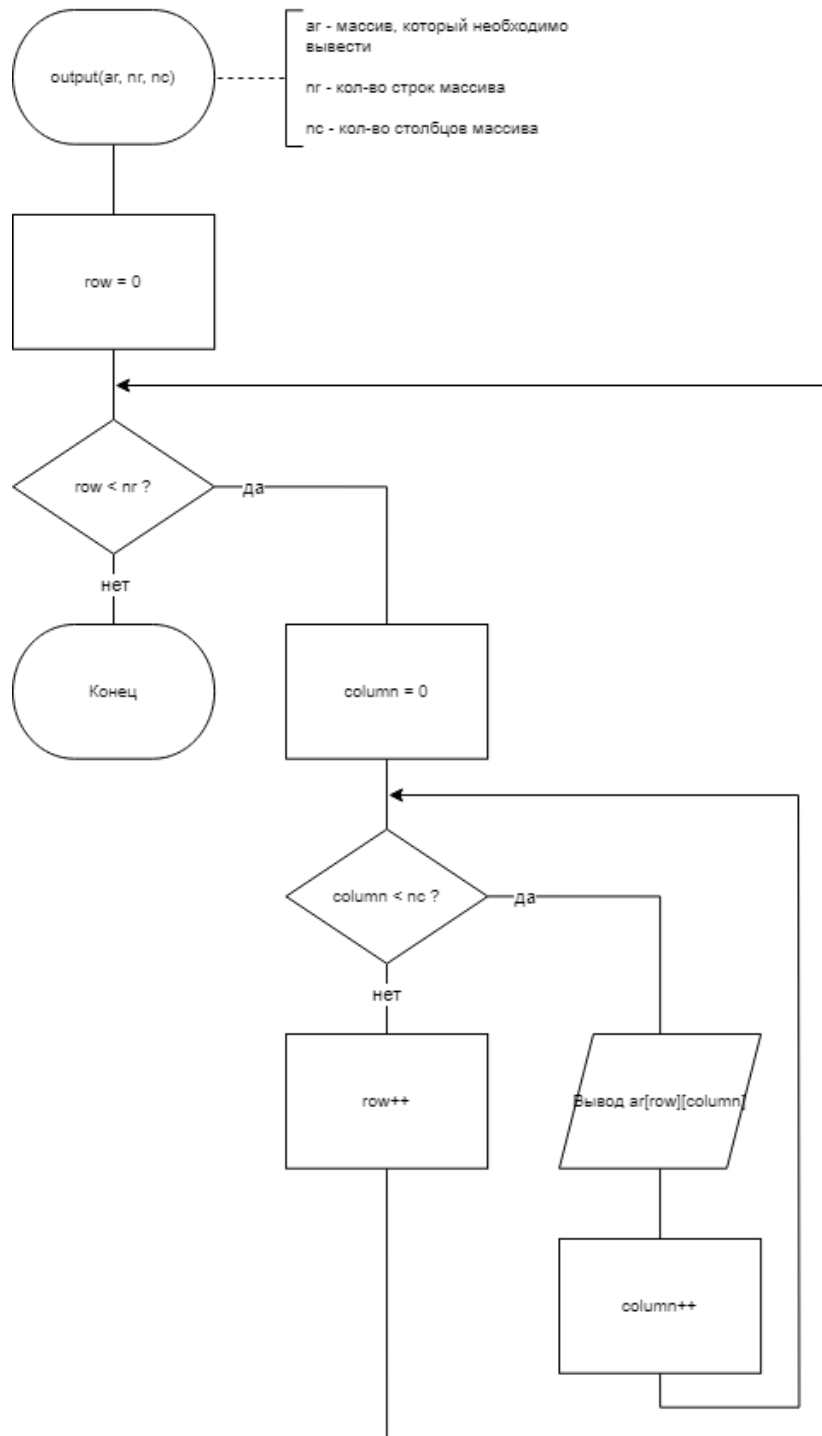
№	Имя переменной	Тип	Назначение
1	column	int	Переменная в цикле, отвечающая за столбец массива.
2	row	int	Переменная в цикле, отвечающая за строку массива.
3	flag	int	Результат проверки выполнения условия

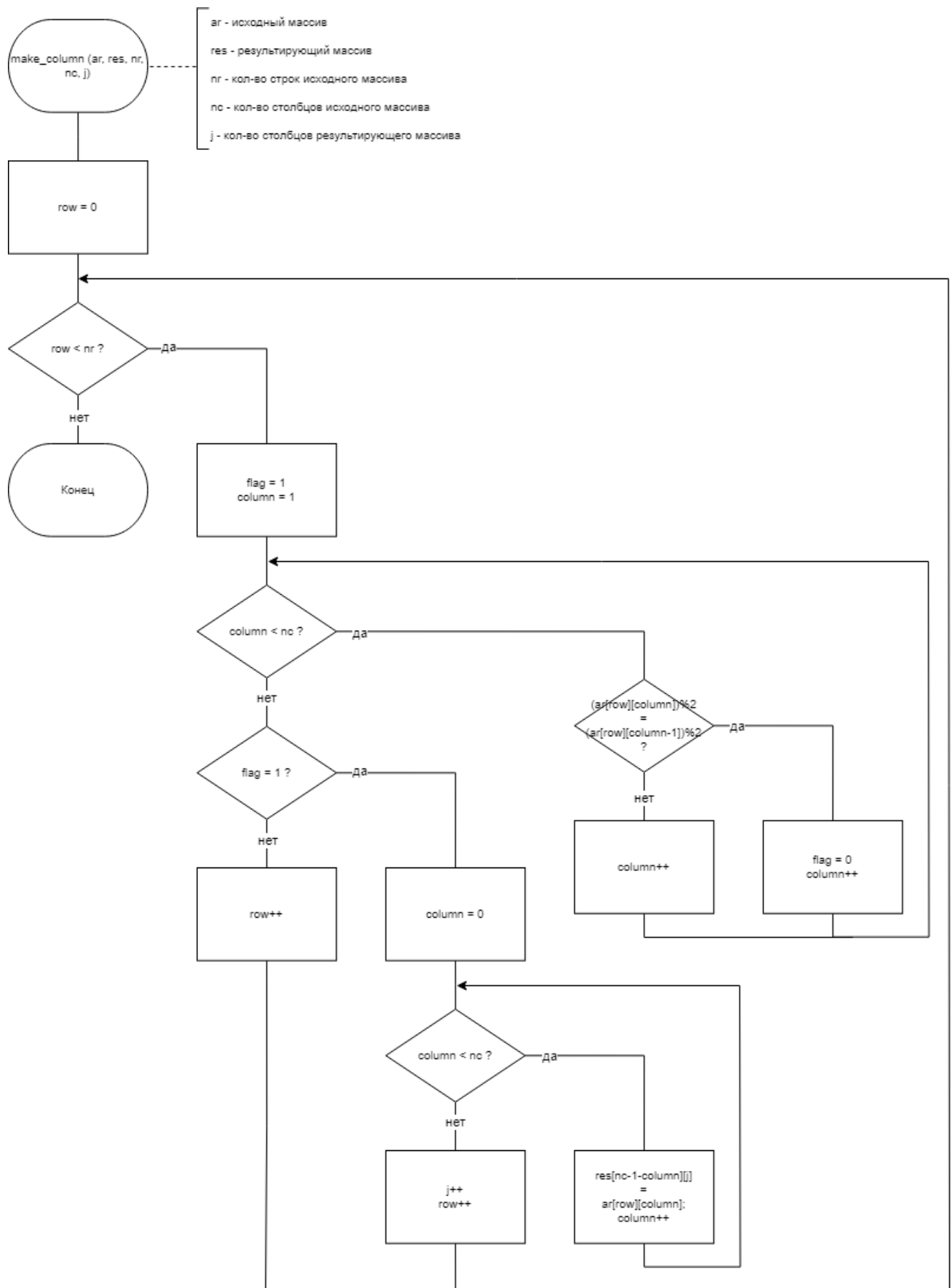
main

№	Имя переменной	Тип	Назначение
1	ar[size][size]	int	Исходный квадратный двумерный массив.
2	res[size][size]	int	Результирующий двумерный массив.
3	column	int	Переменная в цикле, отвечающая за столбец массива.
4	row	int	Переменная в цикле, отвечающая за строку массива.
5	nr	int	Вводимое пользователем значение количества строк
6	nc	int	Вводимое пользователем

			значение количества столбцов
7	j	int	Количество столбцов в результирующем массиве
8	val	int	Значение вводимого элемента массива.

Схема алгоритма





Контрольные примеры

Пример 1:

Исходные данные:

Rows & Columns = 4 4

Original array = 1 2 3 4

2 2 2 2

2 7 8 9

2 4 6 8

Результаты:

Result array = 4 9

3 8

2 7

1 2

Пример 2:

Исходные данные:

Rows & Columns = 3 3

Original array = -5 -3 -1

-2 -5 0

1 2 3

Результаты:

Result array = 0 3

-5 2

-2 1

Пример 3:

Исходные данные:

Rows & Columns = 3 3

Original array = 1 1 1

2 2 2

3 3 3

Результаты:

Result array =

Текст программы

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
#define size 25
```

```
void output(int ar[][size], int nr, int nc){
```

```
    int column,row;
```

```
    for(row=0;row<nr;row++){
```

```
        for(column=0;column<nc;column++){
```

```
            printf("%d ",ar[row][column]);
```

```
        }
```

```
        printf("\n");
```

```
    }
```

```
}
```

```
int make_column(int ar[][size],int res[size][size],int nr,int nc,int j){
```

```
    int column,row,flag;
```

```
    /*Deleting odd rows*/
```

```
    for(row=0;row<nr;row++){
```

```
        flag=1;
```

```
        for(column=1;column<nc;column++){
```

```
            if (((ar[row][column])%2)==((ar[row][column-1])%2)){
```

```
                flag=0;
```

```
        }
```

```

    }
    if (flag==1){
        for(column=0;column<nc;column++){
            res[nc-1-column][j]=ar[row][column];
        }
        j++;
    }
}
return j;
}

```

```

int main(){
    int ar[size][size], column, row, val, nr, nc, res[size][size],j=0;
    printf("Enter the number of Rows and Columns: ");
    scanf("%d %d", &nr, &nc);
    printf("\nEnter array:\n");
    for(row=0; row<nr;row++){
        for(column=0;column<nc;column++){
            scanf("%d", &val);
            ar[row][column]=val;
        }
    }
    printf("\nOriginal array:\n");
    output(ar,nc,nr);
    j=make_column(ar,res,nr,nc,j);
    printf("\nResult array:\n");
    output(res,nc,j);
    return 0;
}

```


Примеры выполнения программы

```
Enter the number of Rows and Columns: 4 4

Enter array:
1 2 3 4
2 2 2 2
2 7 8 9
2 4 6 8

Original array:
1 2 3 4
2 2 2 2
2 7 8 9
2 4 6 8

Result array:
4 9
3 8
2 7
1 2
```

```
Enter the number of Rows and Columns: 3 3

Enter array:
-5 -3 -1
-2 -5 0
1 2 3

Original array:
-5 -3 -1
-2 -5 0
1 2 3

Result array:
0 3
-5 2
-2 1
```

```
Enter the number of Rows and Columns: 3 3

Enter array:
1 1 1
2 2 2
3 3 3

Original array:
1 1 1
2 2 2
3 3 3

Result array:
```

Вывод

В ходе выполнения лабораторной работы были изучены основные принципы создания и использования функций на языке программирования Си. Полученные знания позволят более эффективно использовать функции при разработке программ на языке Си.