



## Отчёт по лабораторной работе № 14

по курсу: 1 фундаментальная информатика

студент группы : М8О-105Б-21 Титеев Рамиль Маратович , № по списку: 23

Адреса www, e-mail, jabber, skype derol.gym@gmail.com

Работа выполнена: “28 ноября 2021г”

Преподаватель: каф. 806 В. К. Титов

Входной контроль знаний с оценкой

Отчёт сдан “    “    20    г., итоговая оценка   

Подпись преподавателя   

1 **Тема:** Вложенные циклы с параметрами. Обход и линеаризация матриц

2 **Цель работы:** Научиться составлять ввода квадратной матрицы и печати в строку ее элементов в заданном порядке следования.

3 **Задание (вариант 17):**

Вывести квадратную матрицу в одну строку, элементы которой будут расположены в следующем порядке:

7	6	5	16
8	1	4	15
9	2	3	14
10	11	12	13

4 **Оборудование (лабораторное):**

ЭВМ   , процессор   , имя узла сети    с ОП    ГБ

НМД    ГБ. Терминал    адрес    . Принтер   

Другие устройства   

*Оборудование ПЭВМ студента, если использовалось:*

Процессор Ryzen4600 @ 6x 3.0GHz, ОП 16384 МБ, НМД    ГБ. Монитор: встроенный

Другие устройства   

5 **Программное обеспечение (лабораторное):**

Операционная система семейства UNIX, наименование:    версия   

Интерпретатор команд:    версия   

Система программирования:    версия   

Редактор текстов:    версия   

Утилиты операционной системы:   

Прикладные системы и программы:   

Местонахождения и имена файлов программ и данных:   

*Программное обеспечение ЭВМ студента, если использовалось:*

Операционная система семейства UNIX, наименование Ubuntu версия 20.04

Интерпретатор команд: bash версия   

Система программирования: C версия   

Редактор текстов: Emacs версия   

Утилиты операционной системы:   

Прикладные системы и программы:   

Местонахождения и имена файлов программ и данных: /usr/bin , а также /bin

- 6 **Идея, метод, алгоритм** решения задачи (в формах: словесной, псевдокода, графической [блок-схема, диаграмма, рисунок, таблица] или формальное описание с пред- и постусловиями)

По условию вывод должен начинаться с центра матрицы. Так как матрица квадратная, её центр будет располагаться в координатах  $(i, j)$ , где  $i = (n+1)/2 - 1$ ,  $j = (n+1)/2 - 1$ . Из этой точки начинает работу цикл, который работает, пока не будет выведено  $n*n$  элементов. Перед началом работы цикла мы выводим центр матрицы, и затем запускаем цикл. В нем работает 4 вложенных цикла `for`, которые выводят наш массив по спирали. Каждые две итерции количество выводимых элементов увеличивается на 1, поэтому каждые два цикла ограничивающий счетчик увеличивается на 1 (переменная `g`).

- 7 **Сценарий выполнения работы** [план работы, первоначальный текст программы в черновике (можно на отдельном листе) и тесты, либо соображения по тестированию].

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

```
#define N 100
```

```
// Инициализация массива
```

```
void InitMas(int mas[N][N], int n){
```

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

```
for (int j = 0; j<n; ++j){
```

```
mas[i][j] = (i+1)*10+j+1;
```

```
}
```

```
}
```

```
}
```

```
// Вывод массива
```

```
void PrintMas(int mas[N][N], int n){
```

```
printf("\nCompleted matrix:\n");
```

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

```
for (int j = 0; j<n; ++j){
```

```
printf("%d ", mas[i][j]);
```

```
}
```

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

```
}
```

```
}
```

```
// Результат
```

```
void Result(int mas[N][N], int n){
```

```
int cnt = 1, i = (n+1)/2 - 1, j = (n+1)/2 - 1, g = 0, c = 0;
```

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

```
printf("%d ", mas[i][j]);
```

```
while (cnt < n*n){
```

```
for(c = ++g; c && cnt < n*n; c--){
```

```
printf("%d ", mas[++i][j]);
```

```
cnt++;
```

```
}
```

```
for(c = g; c && cnt < n*n; c--){
```

```
printf("%d ", mas[i][++j]);
```

```
cnt++;
```

```
}
```

```
for(c = ++g; c && cnt < n*n; c--){
```

```
printf("%d ", mas[--i][j]);
```

```
cnt++;
```

```
}
```

```

        for(c = g; c && cnt < n*n; c--){
            printf("%d ", mas[i][--j]);
            cnt++;
        }

    }
    printf("\n");
}

int main(){
    int mas[N][N], n = 0;
    printf("Input a size of matrix (n>=100): ");
    scanf("%d", &n);
    InitMas(mas, n);
    PrintMas(mas, n);
    Result(mas, n);
    return 0;
}

```

Допущен к выполнению работы. Подпись преподавателя \_\_\_\_\_

#### 8 Распечатка протокола (подклеить листинг окончательного варианта программы с текстовыми примерами, подписанный преподавателем)

```

(base) ramil@ramil:~/labs/lab 14$ cat header.txt
*****
*           Лабораторная работа №14           *
*           Вложенные циклы с параметрами.       *
*           Обход и линеаризация матриц         *
*           Выполнил студент гр. М8О-105-Б       *
*           Титеев Рамиль Маратович             *
*****

```

```

(base) ramil@ramil:~/labs/lab 14$ cat lab14.c
#include <stdio.h>
#define N 100

```

```

// Инициализация массива
void InitMas(int mas[N][N], int n){
    for (int i = 0; i<n; ++i){
        for (int j = 0; j<n; ++j){
            mas[i][j] = (i+1)*10+j+1;
        }
    }
}

```

```

// Вывод массива
void PrintMas(int mas[N][N], int n){
    printf("\nComplited matrix:\n");
    for (int i = 0; i<n; ++i){
        for (int j = 0; j<n; ++j){
            printf("%d ", mas[i][j]);
        }
        printf("\n");
    }
}

```

```

// Результат
void Result(int mas[N][N], int n){
    int cnt = 1, i = (n+1)/2 - 1, j = (n+1)/2 - 1, g = 0, c = 0;
    printf("\nResult:\n");
}

```

```

printf("%d ", mas[i][j]);
while (cnt < n*n){
    for(c = ++g; c && cnt < n*n; c--){
        printf("%d ", mas[++i][j]);
        cnt++;
    }
    for(c = g; c && cnt < n*n; c--){
        printf("%d ", mas[i][++j]);
        cnt++;
    }
    for(c = ++g; c && cnt < n*n; c--){
        printf("%d ", mas[--i][j]);
        cnt++;
    }
    for(c = g; c && cnt < n*n; c--){
        printf("%d ", mas[i][--j]);
        cnt++;
    }
}
printf("\n");
}

```

```

int main(){
    int mas[N][N], n = 0;
    printf("Input a size of matrix (n>=100): ");
    scanf("%d", &n);
    InitMas(mas, n);
    PrintMas(mas, n);
    Result(mas, n);
    return 0;
}

```

(base) ramil@ramil:~/labs/lab 14\$ g++ lab14.c

(base) ramil@ramil:~/labs/lab 14\$ ./a.out

Input a size of matrix (n>=100): 5

Complited matrix:

11 12 13 14 15

21 22 23 24 25

31 32 33 34 35

41 42 43 44 45

51 52 53 54 55

Result:

33 43 44 34 24 23 22 32 42 52 53 54 55 45 35 25 15 14 13 12 11 21 31 41 51

(base) ramil@ramil:~/labs/lab 14\$ ./a.out

Input a size of matrix (n>=100): 8

Complited matrix:

11 12 13 14 15 16 17 18

21 22 23 24 25 26 27 28

31 32 33 34 35 36 37 38

41 42 43 44 45 46 47 48

51 52 53 54 55 56 57 58

61 62 63 64 65 66 67 68

71 72 73 74 75 76 77 78

81 82 83 84 85 86 87 88

Result:

44 54 55 45 35 34 33 43 53 63 64 65 66 56 46 36 26 25 24 23 22 32 42 52 62 72 73 74 75 76 77 67 57 47 37 27 17 16 15 14 13

12 11 21 31 41 51 61 71 81 82 83 84 85 86 87 88 78 68 58 48 38 28 18

(base) ramil@ramil:~/labs/lab 14\$ ./a.out

Input a size of matrix (n>=100): 3

Complited matrix:

11 12 13

21 22 23

31 32 33

Result:

22 32 33 23 13 12 11 21 31

(base) ramil@ramil:~/labs/lab 14\$ ./a.out  
Input a size of matrix (n>=100): 4

Complited matrix:

11 12 13 14  
21 22 23 24  
31 32 33 34  
41 42 43 44

Result:

22 32 33 23 13 12 11 21 31 41 42 43 44 34 24 14

- 9 **Дневник отладки** должен содержать дату и время сеансов отладки, и основные ошибки (ошибки в сценарии и программе, не стандартные операции) и краткие комментарии к ним. В дневнике отладки приводятся сведения об использовании других ЭВМ, существенном участии преподавателя и других лиц в написании и отладке программы.

№	Лаб. или дом.	Дата	Время	Событие	Действие по исправлению	Примечание
1	Дом	28.11.21	13:14	Зациклилась программа, так как забыл прописать счетчик	Дописал счетчик	

- 10 Замечание автора по существу работы \_\_\_\_\_

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

- 11 Выводы \_\_\_\_\_ В этой лабораторной работе я научился представлять матрицу разными способами: в виде матрицы и вектора.

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

Подпись студента \_\_\_\_\_