

# Отчет по лабораторной работе №14

по курсу: 1 фундаментальная информатика

студента группы М80-101Б-21 Тулина Ивана, № по списку: 22

Контакты www, e-mail, icq, skype: i.tulin0107@gmail.com

Работа выполнена: «27» декабря 2021г.

Преподаватель: Титов В. К. каф. 806

Входной контроль знаний с оценкой \_\_\_\_\_

Отчет сдан «    » \_\_\_\_\_ 201 \_\_\_\_ г., итоговая оценка \_\_\_\_\_

Подпись преподавателя \_\_\_\_\_

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

2. **Цель работы:** Составить программу ввода квадратной матрицы и печати в строку всех её элементов в заданном ниже порядке следования (обхода).

3. **Задание (вариант № 22):** Обход и линеаризация квадратной матрицы в следующем порядке

1 12 11 10

2 13 16 9

3 14 15 8

4 5 6 7

4. **Оборудование (лабораторное):**

ЭВМ \_\_\_\_\_ - \_\_\_\_\_, процессор \_\_\_\_\_ - \_\_\_\_\_, имя узла сети \_\_\_\_\_ - \_\_\_\_\_ с ОП \_\_\_\_\_ - \_\_\_\_\_ Мб,

НМД \_\_\_\_\_ Мб. Терминал \_\_\_\_\_ адрес \_\_\_\_\_. Принтер \_\_\_\_\_

Другие устройства \_\_\_\_\_

*Оборудование ПЭВМ студента, если использовалось:*

Процессор Intel Core i5-7300HQ с ОП 7,87 Мб, НМД 15360 Мб. Монитор: встроенный

Другие устройства \_\_\_\_\_

5. **Программное обеспечение (лабораторное):**

Операционная система семейства \_\_\_\_\_ - \_\_\_\_\_ наименование \_\_\_\_\_ - \_\_\_\_\_ версия \_\_\_\_\_ - \_\_\_\_\_, интерпретатор команд \_\_\_\_\_ версия \_\_\_\_\_

Система программирования \_\_\_\_\_ версия \_\_\_\_\_

Редактор текстов \_\_\_\_\_ версия \_\_\_\_\_

Утилиты операционной системы \_\_\_\_\_

Прикладные системы и программы: \_\_\_\_\_

Местонахождение и имена файлов программ и данных \_\_\_\_\_

*Программное обеспечение ЭВМ студента, если использовалось:*

Операционная система семейства UNIX, наименование Ubuntu версия 20.04.3 LTS

интерпретатор команд bash версия \_\_\_\_\_.

Система программирования \_\_\_\_\_ версия \_\_\_\_\_

Редактор текстов Emacs версия 3.22.30

Утилиты операционной системы \_\_\_\_\_ - \_\_\_\_\_

Прикладные системы и программы \_\_\_\_\_ - \_\_\_\_\_

Местонахождение и имена файлов программ и данных на домашнем компьютере \_\_\_\_\_ - \_\_\_\_\_

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

На вход программы поступает целое неотрицательное число – порядок квадратной матрицы.

В цикле заполняем матрицу, вычисляя каждый элемент по формуле:  $a[i][j] = (i+1) * 10 + j + 1$ , где  $a[i][j]$  – элемент,  $i$  – номер строки матрицы,  $j$  – номер столбца матрицы.

Процесс обхода матрицы разбиваем на процесс обхода каждого отдельного кольца с номером  $k$   
Количество колец вычисляется по формуле  $U = n/2 + n\%2$ , где  $U$  – количество колец,  $n$  – порядок матрицы

Обходим каждое кольцо, начиная с левого верхнего угла, используя для каждого четыре цикла

- в первом цикле выводятся элементы  $a[i][j]$ , где  $j=k$ ,  $i$  меняется от  $k$  до  $p+1=n-k$  (элементы левой грани кольца)
- во втором цикле выводятся элементы  $a[i][j]$ , где  $i=p-n-k-1$ ,  $j$  меняется от  $k+1$  до  $p$  (элементы нижней грани кольца)
- в третьем цикле выводятся элементы  $a[i][j]$ , где  $j=p$ ,  $i$  меняется от  $p$  до  $k$  (элементы правой грани кольца)
- в четвертом цикле выводятся элементы  $a[i][j]$ , где  $i=k$ ,  $j$  меняется от  $p$  до  $k$  (элементы верхней грани кольца)

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

```
#include<stdio.h>
int main(){int n,i,j,k,p,a[9][9];

printf("Input n=");
scanf("%d", &n);

printf("Source matrix:\n");
for(i=0;i<n;++i) {for(j=0;j<n;++j)
{a[i][j]=(i+1)*10+j+1;
printf("%4d", a[i][j]);}
printf("\n");}

printf("Result vector\n");
for(k=0; k<(n/2+n%2);++k) // цикл по обходу колец
{p=n-k-1;
for(i=k;i<p+1;++i) // обход левой грани кольца
{j=k;printf("%4d", a[i][j]);}
for(j=k+1;j<p;++j) //обход нижней грани кольца
{i=p;printf("%4d", a[i][j]);}
for(i=p;i>k;--i) //обход правой грани кольца
{j=p;printf("%4d", a[i][j]);}
for(j=p;j>k;--j) //обход верхней грани кольца
{i=k;printf("%4d", a[i][j]);}}
printf("\n");
return 0; }
```

Пункты 1-7 отчета составляются строго до начала лабораторной работы.

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

**8. Распечатка протокола** (подклеить листинг окончательного варианта программы с тестовыми примерами, подписанный преподавателем).

```
yusayu@YS:~/Рабочий стол/cppProjects$ cat head
```

```
*****
*
*      Лабораторная работа №14      *
*      Вложенные циклы с параметрами *
*      Обход и линеаризация матриц   *
*      Выполнил: Тулин Иван Денисович *
*      (номер по списку: 22)          *
*      Группа: М8О-101Б-21            *
*
*****
```

```
yusayu@YS:~/Рабочий стол/cppProjects$ cat lab14.c
```

```
#include<stdio.h>
int main(){int n,i,j,k,p,a[9][9];
printf("Input n=");
scanf("%d", &n);

printf("Source matrix:\n");
for(i=0;i<n;++i) {for(j=0;j<n;++j)
{a[i][j]=(i+1)*10+j+1;
printf("%4d", a[i][j]);}
printf("\n");}
```

```
printf("Result vector\n");
for(k=0; k<(n/2+n%2);++k)
{p=n-k-1;
for(i=k;i<p+1;++i)
{j=k;printf("%4d", a[i][j]);}
for(j=k+1;j<p+1;j++)
{i=p;printf("%4d", a[i][j]);}
for(i=p;i>k;--i)
{j=p;printf("%4d", a[i][j]);}
for(j=p;j>k;--j)
{i=k;printf("%4d", a[i][j]);}}
printf("\n");
return 0; }
```

```
yusayu@YS:~/Рабочий стол/cppProjects$ gcc -o lab14.out lab14.c
```

```
yusayu@YS:~/Рабочий стол/cppProjects$ ./lab14.out
```

```
Input n=5
```

```
Source 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 vector
```

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

```
yusayu@YS:~/Рабочий стол/cppProjects$ ./lab14.out
```

```
Input n=4
```

```
Source matrix:
```

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

```
Result vector
```

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

```
yusayu@YS:~/Рабочий стол/cppProjects$ ./lab14.out
```

```
Input n=1
```

Source matrix:

11

Result vector

11

9. **Дневник отладки** должен содержать дату и время сеансов отладки, и основные события (ошибки в сценарии и программе, нестандартные ситуации) 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

10. **Замечания автора** по существу работы \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

11. **Выводы**  
В ходе работы я научился составлять программы обхода и линеаризации квадратных матриц \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

Недочёты при выполнении задания могут быть устранены следующим образом: \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

Подпись студента \_\_\_\_\_