

Отчет по лабораторной работе № 14 по курсу “Фундаментальная информатика”

Студент группы М80-109Б-22 Серый Никита Олегович, № по списку 16

Контакты:

email — nikita.seryj@mail.ru

telegram - @hukumkas

Работа выполнена: «24» декабря 2022г.

Преподаватель: каф. 806 Сысоев Максим Алексеевич

Отчет сдан «24» декабря 2022 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 порядке следования(обхода).

3. **Задание (вариант № 26):**

26

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

4. **Оборудование (студента):**

Процессор *AMD Ryzen 5 5500U @ 6x 2.1GH* с ОП *15360* Мб, НМД (?) *512* Гб. Монитор *1920x1080*

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

Операционная система семейства: *linux*, наименование: *mint*, версия *21 Cinnamon*
интерпретатор команд: *bash* версия *4.4.19*.

Система программирования -- версия --, редактор текстов *emacs* версия *25.2.2*

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)

1. Объявляем размер матрицы, саму матрицу и используемые переменные.
2. Заполним матрицу значениями с клавиатуры.
3. Выведем её на экране для наглядности
4. Запишем стартовое положение (координаты)
5. Напишем случаи для обхода через switch-case
6. Выведем результат в линейном виде на экране

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

Входные данные	Выходные данные	Описание тестируемого случая
3 Input matrix: 11 12 13	Linear form: 19 18 17 14 11 12 13 16 15	Все верно

14 15 16 17 18 19		
4 Input matrix: 11 12 13 14 21 22 23 24 31 32 33 34 41 42 43 44	Linear form: 44 43 42 41 31 21 11 12 13 14 24 34 33 32 22 23	Все верно
5 Input matrix: 1 2 3 4 5 11 12 13 14 15 21 22 23 24 25 31 32 33 34 35 41 42 43 44 45	Linear form: 45 44 43 42 41 31 21 11 1 2 3 4 5 15 25 35 34 33 32 22 12 13 14 24 23	Все верно

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

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

```
#define MAX_N 7
```

```
/* программа реализована с помощью switch-case для большей наглядности */
```

```
typedef enum _type_direction // объявим перечислимый тип состояний, как в 11 лабе
```

```
{
    LEFT = 0,
    UP,
    RIGHT,
    DOWN
} type_direction;
```

```
int main()
```

```
{
    // в теле функции задали постоянный размер матрицы, ввели используемые переменные
    int square_size, matrix[MAX_N][MAX_N], i, j, k, leftIndex, rightIndex, upIndex, downIndex;
    type_direction dir;
```

```
    // 1 - булевское значение. 'пока истина', т.е. пока программа не завершена, выполняем цикл
    while (1)
```

```
{
    printf("\nPrint square size here (1 - %d. Break - 0): ", MAX_N);
    scanf("%d", &square_size);
```

```
    if (square_size == 0){
break;
    }
```

```
    if (square_size < 1 || square_size > MAX_N){
        printf("\nRestricted value of size!\n");
        continue;
    }
```

```
    // поэлементно заполним матрицу нашими значениями
    for (i = 0; i < square_size; i++){
        for (j = 0; j < square_size; j++){
            scanf("%d", &matrix[i][j]);
        }
    }
```

```

// выведем поэлементно нашу матрицу на экране
printf("Input matrix:\n");
for (i = 0; i < square_size; i++){
    for (j = 0; j < square_size; j++){
printf("%d ", matrix[i][j]);
    }

    printf("\n"); // выводим символ новой строки в конце первого цикла
}

```

// по условию варианта направление обхода начинается из правого нижнего угла, затем влево и по краю в улиточной форме идёт вовнутрь

```

dir = LEFT;
leftIndex = 0; // левый индекс нулевой (что очевидно)
rightIndex = square_size - 1; // правый индекс будет меньше, чем размер матрицы, на 1 (счит
здесь от нуля)
upIndex = 0;
downIndex = square_size - 1;
i = square_size - 1;
j = square_size - 1;

```

```

// выведем линейный вид матрицы
printf("Linear form: ");
for (k = 0; k < square_size * square_size; k++) {
    printf("%d ", matrix[i][j]);
}

```

```

// дальше идут условия, когда меняем направление при обходе матрицы
switch (dir){

```

```

    case LEFT:
    {
        if (j > leftIndex) {
            j--;
        }
        else
        {
            i--;
            downIndex--;
            dir = UP;
        }
    }
    break;

```

```

    case UP:
    {
        if (i > upIndex){
            i--;
        }
        else
        {
            j++;
            leftIndex++;
            dir = RIGHT;
        }
    }
    break;

```

```

    case RIGHT:
    {

```

```

        if (j < rightIndex){
            j++;
        }
        else
        {
            i++;
            upIndex++;
            dir = DOWN;
        }
    }
    break;

case DOWN:
{
    if (i < downIndex){
        i++;
    }
    else
    {
        j--;
        rightIndex--;
        dir = LEFT;
    }
}
break;
    }
}

printf("\n");
}

return 0;
}

```

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
1	дом	24.12.2022	13:45	Пытался решить задачу как в методичке	Отложил решение	Последовательность действий ясна
2	дом	24.12.2022	20:40	Решил задачу без использования идеи остатков от деления	Написал стадии переходов при обходе матрицы	Получилось!

10. Замечания автора по существу работы

Не вник в объяснение из методички. Непонятно, что означает массив $\text{int dir}[4] = \{1,0,-1,1\}$ и как с ним взаимодействовать.

11. Выводы

Работа оказалась непростой. Думаю, позже до меня дойдёт более лаконичное решение данной задачи. А пока я приступаю к следующим лабораторным! Время идёт.

Недочёты при выполнении задания могут быть устранены следующим образом: --

Подпись студента _____