

Отчет по лабораторной работе № 14 по курсу Архитектура компьютера и информационных сетей

Студент группы М8О-103Б-22 Клименко Виталий Максимович, № по списку 11

Контакты www, e-mail, icq, skype vitalikklimenko96@gmail.com

Работа выполнена: 2022 г.

Преподаватель: доцент Никулин С.П.

Входной контроль знаний с оценкой _____

Отчет сдан « » _____ 202__ г., итоговая оценка _____

Подпись преподавателя _____

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

- Цель работы:** Научиться работать со вложенными циклами

- Задание (вариант № 13):** Составить программу ввода квадратной матрицы и печати в строку всех в заданном ниже порядке обхода

- Оборудование (лабораторное):**
ЭВМ _____, процессор _____, имя узла сети _____ с ОП _____ Мб,
НМД _____ Мб. Терминал _____ адрес _____. Принтер _____
Другие устройства _____
Оборудование ПЭВМ студента, если использовалось:
Процессор Intel 4x 3.5GHz _____ с ОП 16 Гб _____ НМД HDD 200 Гб _____. Монитор Встроенный 1920x1080
Другие устройства Touchpad Synaptics _____
- 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:
интерпретатор команд bash _____ версия 5.1.16
Операционная система семейства UNIX _____, наименование Pop!_OS _____ версия 22.04 jammy
Система программирования GNU Compiler Collection (GCC) _____ версия 11.2.0
Редактор текстов Visual Studio Code _____ версия 1.72.2
Утилиты операционной системы _____
Прикладные системы и программы gcc, gdb _____
Местонахождение и имена файлов программ и данных на домашнем компьютере _____

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

Сделать программу, в которой вводится матрица вводимого размера, с помощью функции вывести матрицу в порядке заданного варианта. В функции вывода использовать вложенные циклы

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

- 1) Написать ввод матрицы
- 2) Написать основную функцию вывода и функцию для дебага

Тесты:

1)

1 2 3

4 5 6

7 8 9 -> 3 5 7 4 2 8 6 9 1

2)

15 13 7 1

14 6 2 10

5 3 9 11

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

3)

1 2 3 4 5 6 7

8 9 10 11 12 13 14

15 16 17 18 19 20 21

22 23 24 25 26 27 28

29 30 31 32 33 34 35

36 37 38 39 40 41 42

43 44 45 46 47 48 49 -> 7 13 19 25 31 37 43 36 30 24 18 12 6 44 38 32 26 20 14 21 27 33 39 45 5 11 17 23 29

22 16 10 4 46 40 34 28 35 41 47 3 9 15 8 2 48 42 49 1

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)

```
vital@vitos-hp16 MINGW64 /c/important/docs/mai/labs/l14 (master)
$ cat makefile
CC = gcc
CFLAGS = -Wall -Wextra

main:
    $(CC) $(CFLAGS) -o main.out main.c
    ./main.out

vital@vitos-hp16 MINGW64 /c/important/docs/mai/labs/l14 (master)
$ cat main
main.c    main.out

vital@vitos-hp16 MINGW64 /c/important/docs/mai/labs/l14 (master)
$ cat main.c
#include <stdio.h>
#include <stdlib.h>

void printmd(long long** m, long long n) {
    for (long long i = 0; i < n; ++i) {
        for (long long j = 0; j < n; ++j) {
            printf("%lld ", m[i][j]);
        }
        printf("\n");
    }
}

void printm(long long** m, long long n) {
    long long j1 = n - 1, j2 = 1;

    while (1) {
        if (j1 > -1) {
            long long i = 0, j = j1;
            for (; j > -1; --j) {
                printf("%lld ", m[i][j]);
                ++i;
            }
            i -= 2;
            ++j;
            for (; i > -1; --i) {
                printf("%lld ", m[i][j]);
                ++j;
            }
        }
        if (j2 < n) {
            long long i = n - 1, j = j2;
            for (; j < n; ++j) {
                printf("%lld ", m[i][j]);
                --i;
            }
            i += 2;
            --j;
            for (; i < n; ++i) {
                printf("%lld ", m[i][j]);
                --j;
            }
        }
        j1 -= 2;
        j2 += 2;
        if (j1 < 0 && j2 > n - 1) {
            printf("\n");
            break;
        }
    }
}

int main() {
    long long n;
    scanf("%lld", &n);

    long long** m = (long long**) malloc(n * sizeof(long long*));
    for (long long i = 0; i < n; ++i) {
        m[i] = (long long*) malloc(sizeof(long long) * n);
    }

    for (long long i = 0; i < n; ++i) {
        for (long long j = 0; j < n; ++j) {
            scanf("%lld", &m[i][j]);
        }
    }
}
```

```

    }

    // printf("\n");
    // printmd(m, n);
    printf("\n");
    printm(m, n);

    return 0;
}

vital@vitos-hp16 MINGW64 /c/important/docs/mai/labs/l14 (master)
$ make
gcc -Wall -Wextra -o main.out main.c
./main.out
1
1
1

vital@vitos-hp16 MINGW64 /c/important/docs/mai/labs/l14 (master)
$ make
gcc -Wall -Wextra -o main.out main.c
./main.out
2
1 2
3 4

2 3 1 4

vital@vitos-hp16 MINGW64 /c/important/docs/mai/labs/l14 (master)
$ make
gcc -Wall -Wextra -o main.out main.c
./main.out
3
1 2 3
4 5 6
7 8 9

3 5 7 4 2 8 6 9 1

vital@vitos-hp16 MINGW64 /c/important/docs/mai/labs/l14 (master)
$ make
gcc -Wall -Wextra -o main.out main.c
./main.out
4
15 13 7 1
14 6 2 10
5 3 9 11
4 8 12 16

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

vital@vitos-hp16 MINGW64 /c/important/docs/mai/labs/l14 (master)
$ make
gcc -Wall -Wextra -o main.out main.c
./main.out
5
1 2 3 4 5
6 7 8 9 10
11 12 13 14 15
16 17 18 19 20
21 22 23 24 25

5 9 13 17 21 16 12 8 4 22 18 14 10 15 19 23 3 7 11 6 2 24 20 25 1

vital@vitos-hp16 MINGW64 /c/important/docs/mai/labs/l14 (master)
$ make
gcc -Wall -Wextra -o main.out main.c
./main.out
7
1 2 3 4 5 6 7
8 9 10 11 12 13 14
15 16 17 18 19 20 21
22 23 24 25 26 27 28
29 30 31 32 33 34 35
36 37 38 39 40 41 42
43 44 45 46 47 48 49

7 13 19 25 31 37 43 36 30 24 18 12 6 44 38 32 26 20 14 21 27 33 39 45 5 11 17 23 29 22 16 10 4 46 40 34 28
35 41 47 3 9 15 8 2 48 42 49 1

vital@vitos-hp16 MINGW64 /c/important/docs/mai/labs/l14 (master)
$ make
gcc -Wall -Wextra -o main.out main.c
./main.out

```

5

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

15 24 33 42 51 41 32 23 14 52 43 34 25 35 44 53 13 22 31 21 12 54 45 55 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 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 данной лабораторной работы я научился задавать двойные массивы, выполнять операции над ними (динамически их задавать, выполнять индексирование, передавать в функции). Также я больше узнал про циклы `for` и `while` и их параметрические задание

Недочёты при выполнении задания могут быть устранены следующим образом: _____

Подпись студента _____