



Отчет по лабораторной работе № 14 по курсу Фундаментальная информатика

Студент группы М8О-104Б-22 Ляпин Иван Алексеевич, № по списку 00

Контакты www, e-mail, icq, skype shad0w2020@mail.ru

Работа выполнена: « 12 » ноября 2022 г.

Преподаватель: асп. каф. 806 Потенко М.А.

Входной контроль знаний с оценкой _____

Отчет сдан « » _____ 202__ 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 ее элементов в заданном порядке обхода _____

3. **Задание (вариант № 20):** Вывести элементы матрицы змейкой, начиная с правого нижнего угла _____

4. **Оборудование (лабораторное):**

ЭВМ _____, процессор _____, имя узла сети _____ с ОП _____ Мб,
НМД _____ Мб. Терминал _____ адрес _____. Принтер _____
Другие устройства _____

Оборудование ПЭВМ студента, если использовалось:

Процессор Apple M1 Pro с ОП 16384 Мб, НМД 524 288 Мб. Монитор Liquid Retina XDR

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 macOS, наименование macOS Monterey версия 12.3

интерпретатор команд zsh версия 2.12.5

Система программирования C версия _____

Редактор текстов nano версия _____

Утилиты операционной системы Терминал _____

Прикладные системы и программы Xcode

Местонахождение и имена файлов программ и данных /Users/ivan/Desktop _____

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

Первым делом будем считать порядок матрицы.

Создаем двумерный массив(матрицу) размера $n * n$, где n - порядок матрицы.

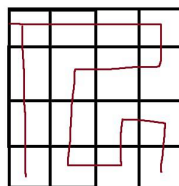
С помощью клавиатуры заполняем значения нашей матрицы в цикле.

Так как нам нужно вывести все эл-ты змейкой, начиная с правого нижнего угла, то можно поступить так:

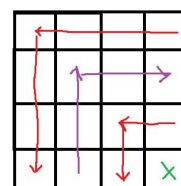
Выводим элемент из угла, затем выводим элементы "углами" справа-налево, слева-направо.

20			
13	12	11	10
14	7	8	9
15	6	3	2
16	5	4	1

порядок вывода



алгоритм вывода



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

- 1) Введем переменную "n", отвечающую за порядок матрицы, данное значение мы считываем с клавиатуры.
- 2) Если $n = 0$, то завершим работу программы.
- 3) Создадим матрицу a, размера $n * n$.
- 4) Заполним нашу матрицу с помощью клавиатуры через два цикла, так как наш массив двумерный.
- 5) Пусть "num_elem" - кол-во элементов в матрице ($n * n$), delta_Str_Col - параметр, отвечающий за уменьшение строк и столбцов (изначально = 0), count_words - число уже выведенных слов из матрицы.
- 6) Выведем самый последний элемент матрицы, то есть $a[n-1][n-1]$.
- 7) Запускаем цикл пока $count_words < num_elem$.
- 8) Выводим элементы $a[n - 2 - delta_Str_Col][k]$ в цикле, от $k = n - 1$ пока $k \geq n - 2 - delta_Str_Col$, увеличиваем count_words с каждым выводом на единицу.
- 9) Выводим элементы $a[k][n - 2 - delta_Str_Col]$ в цикле, от $k = n - 1 - delta_Str_Col$ пока $k \leq n - 1$, увеличиваем count_words с каждым выводом на единицу.
- 10) Увеличиваем параметр delta_Str_Col на единицу, чтобы перейти к следующему углу, только теперь мы выводим слева-направо для сохранения порядка, указанного в варианте.
- 11) Выполняем проверку $count_words == num_elem$, если да - завершаем цикл, нет - продолжаем.
- 12) Выводим элементы $a[k][n - 2 - delta_Str_Col]$ в цикле, от $k = n - 1$ пока $k \geq n - 2 - delta_Str_Col$, увеличиваем count_words с каждым выводом на единицу.
- 13) Выводим элементы $a[n - 2 - delta_Str_Col][k]$ в цикле, от $k = n - 1 - delta_Str_Col$ пока $k \leq n - 1$, увеличиваем count_words с каждым выводом на единицу.
- 14) Повторяем 10-ый шаг.
- 15) После выполнения цикла завершаем работу программы.

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)

```
1 /*
2     Ляпин Иван М80-1045-22
3     Лабораторная работа W 14
4     Вариант: 20
5
6     */
7
8
9 #include <math.h>
10 #include <ctype.h>
11 #include <stdio.h>
12 #include <stdlib.h>
13 #include <string.h>
14 #include <ctype.h>
15 #include <string.h>
16 #include <stdbool.h>
17
18
19 int main(int argc, const char * argv[])
20 {
21     int n; // порядок матрицы
22     printf("Введите порядок матрицы\n");
23     scanf("%d", &n); // считываем значение
24     if(n == 0) {
25         return 0;
26     }
27     int a[n][n]; // двумерный массив(матрица)
28     printf("Введите элементы введенной матрицы\n");
29     for (int i = 0; i < n; i++) {
30         // запускаем цикл по заполнению элементов в матрицу с клавиатуры
31         for (int j = 0; j < n; j++) {
32             scanf("%d", &a[i][j]);
33         }
34     }
35
36     int num_elem = n * n; // кол-во элементов в матрице
37     int delta_Str_Col = 0; // вводим параметр, отвечающий за уменьшение строк и столбцов
38     int count_words = 1; // число уже выведенных(на момент в цикле) элементов на экран спиралью
39
40     printf("%3d\t", a[n-1][n-1]); // выводим элемент из правого нижнего угла
41
42     while(count_words < num_elem) { // цикл вывода будет выполняться, пока не будут выведены все эл-ты матрицы
43         // выводим угол при движении вправо
44         for(int k = n - 1; k >= n - 2 - delta_Str_Col; k--) {
45             printf("%3d\t", a[n - 2 - delta_Str_Col][k]);
46             count_words++;
47         }
48         for(int k = n - 1 - delta_Str_Col; k <= n - 1; k++) {
49             printf("%3d\t", a[k][n - 2 - delta_Str_Col]);
50             count_words++;
51         }
52         delta_Str_Col++; //увеличиваем параметр, так как только что закончился вывод угла справа-налево, последующий вывод
53             //будет иметь большое кол-во элементов за счет нахождения на один столбец и строку меньше
54
55         //проверка, чтобы не вывести ненужный "мусор"
56         if (count_words == num_elem) {
57             break;
58         }
59         // выводим угол при движении влево
60         for (int k = n - 1; k >= n - 2 - delta_Str_Col; k--) {
61             printf("%3d\t", a[k][n - 2 - delta_Str_Col]);
62             count_words++;
63         }
64         for (int k = n - 1 - delta_Str_Col; k <= n - 1; k++) {
65             printf("%3d\t", a[n - 2 - delta_Str_Col][k]);
66             count_words++;
67         }
68         delta_Str_Col++; // увеличиваем параметр, закончили вывод угла слева-направо
69
70
71     }
72
73     return 0;
74 }
75
76 }
```

Тест:

Введите порядок матрицы

7

Введите элементы введенной матрицы

43 44 45 46 47 48 49

42 31 30 29 28 27 26

41 32 21 22 23 24 25

40 33 20 13 12 11 10

39 34 19 14 7 8 9

38 35 18 15 6 3 2

37 36 17 16 5 4 1

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	Program ended with exit code: 0																				

Введите порядок матрицы

0

Program ended with exit code: 0

Достаточно будем продемонстрировать результат работы для матрицы $7 * 7$, так как матрицы порядка < 7 , являются её минорами, тем самым очевидно, что программа будет работать исправно, также проведем тест с введенным порядком, равным нулю.

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 данной лабораторной работе я научился работать с двумерными массивами, выводить элементы в различном порядке. Также я закрепил свои знания циклов при работе с ними, научился выполнять их с указанными параметрами. Матрицы являются базовой и важнейшей частью линейной алгебры. Матрицы часто используются в программировании, например в 3D-моделировании или гейм-девелопинге. Также матрицы значительно облегчают работы с программами, где необходимо взаимодействовать с тысячами переменных. Матрицы широко задействованы в нейронных сетях. Например можно "записывать" изображение в виде матрицы, где её значения будут варьироваться в зависимости от значения оттенка цвета, тем самым в последствии программа сможет различить, что изображено на картинке.

Недочёты при выполнении задания могут быть устранены следующим образом: _____

Подпись студента

A handwritten signature in black ink, consisting of several loops and a long horizontal stroke, positioned above a horizontal line.