

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

Студент группы М80-109Б-22 Любарский Иван Владимирович, № по списку 8

Контакты ivanred289@gmail.com, @rMeDGranD

Работа выполнена: «18» декабря 2022г.

Преподаватель: каф. 806 Сысоев Максим Алексеевич

Отчет сдан « » _____ 20__ г., итоговая оценка _____

Подпись преподавателя _____

1. **Тема:** Вложенные циклы с параметрами. Обход и линеаризация матриц.
2. **Цель работы:** Составить программу на языке Си для анализа и обработки матриц.
3. **Задание:** Вывод матрицы «Улиткой» с нижнего левого края. (Вариант 24)
4. **Оборудование:**

Процессор *Intel Core i5-4210U @ 4x 1.7GH* с ОП *15873 Мб*, НМД *512 Гб*. Монитор *1600x900*

5. Программное обеспечение:

Операционная система семейства: *linux*, наименование: *ubuntu*, версия *22.04 LTS*

интерпретатор команд: *bash* версия *4.4.19*.

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

Утилиты операционной системы --

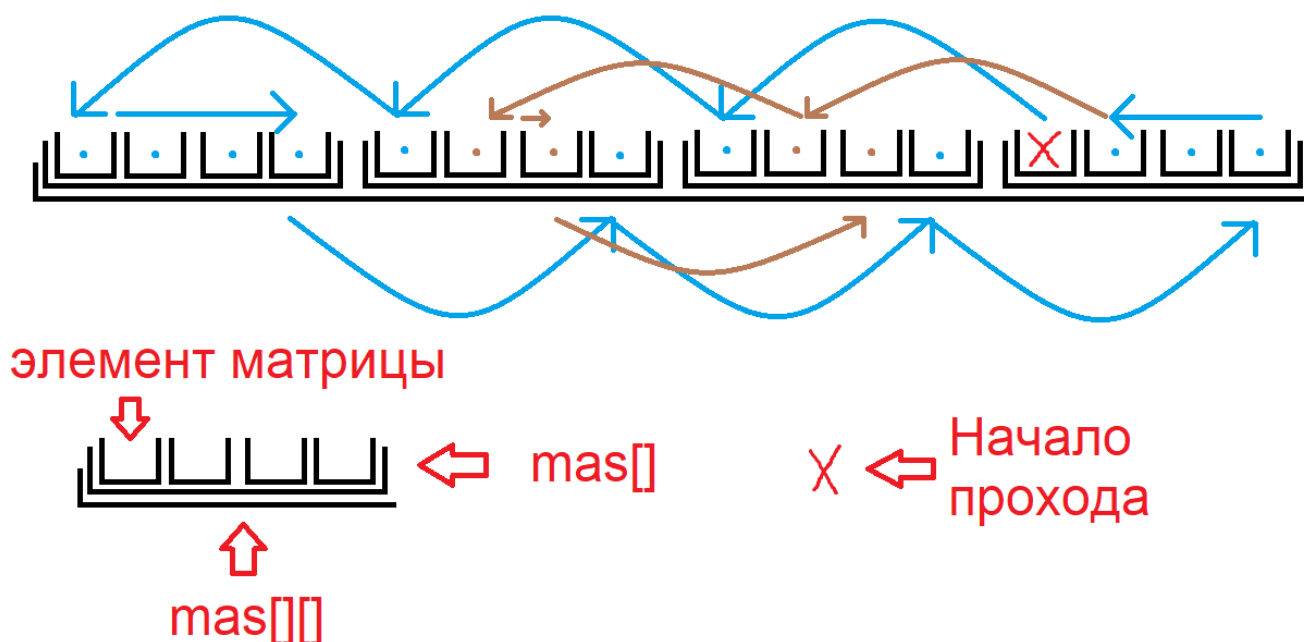
Прикладные системы и программы --

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)

Всю матрицу в языке Си можно представить в виде двумерного массива (Указателем на указатели). Поэтому ее можно представить в виде прямого множества с , возможными, повторяющимися элементами. Индексация этого множества будет идти от 0 до квадрата ее размерности. Все прямую можно разделить на части из количества элементов равного размерности.

Тогда, если идти по алгоритму вывода можно заметить, что происходят два главных набора действий с множеством — это переход «Влево» и «Вправо». Осталось только реализовать эти действия. Описание этого алгоритма на матрице 4 на 4 можно увидеть на следующем изображении:



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

План работы программы:

1. Считать матрицу
2. Произвести проход Влево
3. произвести проход Вправо
4. Если количество выведенных элементов равно количеству элементов матрицы — закончить программу.

Входные данные	Выходные данные	Описание тестируемого случая
1	1	Вывод тривиального случая из матрицы 1 на 1
1 2 3 4	3 1 2 4	Вывод матрицы 2 на 2
1 2 3 4 5 6 7 8 9	7 4 1 2 3 6 9 8 5	Вывод матрицы 3 на 3
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16	13 9 5 1 2 3 4 8 12 16 15 14 10 6 7 11	Вывод матрицы 4 на 4

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

```
#include <stdio.h>
#define N 3 //Размер матрицы

long long mas[N][N]; //Матрица

int main() {

    long long start = N * (N - 1); //Начальный элемент вывода матрицы
    long long kol = 0; //Счетчик выведенных элементов

    for (int i = 0; i < N * N; i++) { scanf("%lli", &mas[i/N][i%N]); } //Считывание матрицы

    while (1){
        while (start / N != start % N) { printf("%lli ", mas[start / N][start % N]); start -= N; kol++; if (kol == N * N) break; }
        // { Вывод; Изменение start; Инкремент kol; Проверка на конец матрицы; }
        if (kol == N * N) { break; }
        while (start % N != N - start / N - 1) { printf("%lli ", mas[start / N][start % N]); start++; kol++; if (kol == N * N)
            break; } //Конец прохода "Влево"
        if (kol == N * N) { break; }

        while (start % N != start / N) { printf("%lli ", mas[start / N][start % N]); start += N; kol++; if (kol == N * N) break; }
        if (kol == N * N) { break; }
        while (start % N != N - start / N) { printf("%lli ", mas[start / N][start % N]); start--; kol++; if (kol == N * N) break; }
        //Конец прохода "Вправо"
        if (kol == N * N) { break; }
    }

    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

10. Замечания автора по существу работы

11. Выводы

Данная работа помогла лучше понять работу указателей из-за использования указателей(массивов) и указателей на указатели. Также важно можно отметить необходимость разбить алгоритм выполнения данной работы на части для более легкой работы, что может очень сильно понадобиться в будущем.

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

Подпись студента _____