Міністерство освіти і науки України

Національний технічний університет

«Харківський політехнічний інститут»

Кафедра «Обчислювальна техніка та програмування»

**ЗВІТ**

Про виконання лабораторної роботи № 11

«Вступ до показчиків»

Керівник: викладач

Бульба С. С.

Виконавець: студент гр. КІТ-120в

Стегній Б. В.

Харків 2020

**Лабораторна робота № 11. Вступ до показчиків**

**1 Вимоги**

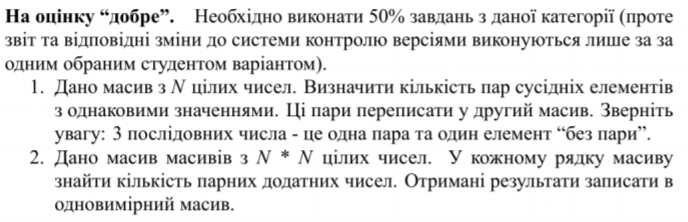
* 1. **Розробник**

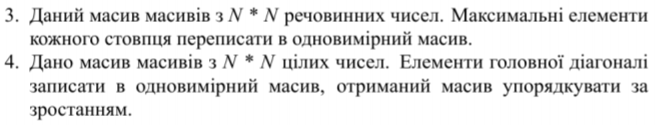
• Стегній Богдан Володимирович;

• Студент групи КІТ-120в;

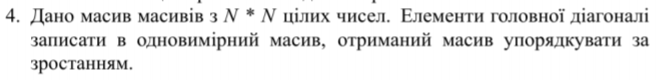
• 12-груд-2020.

* 1. **Загальне завдання**





**1.3 Індивідуальне завдання**



1. **Виконання роботи**
   1. Створив директорію lab11 у репозиторії Programing-repo.



Рисунок 1 – Створення директорії



Рисунок 2 – створення директорій

* 1. Заніс потрібні файлі до директорій для того, щоб можна було працювати.

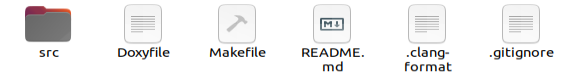


Рисунок 3 – потрібні файли для роботи

* 1. Написав код для четвертого завдання з використанням фунцій та показчиків.

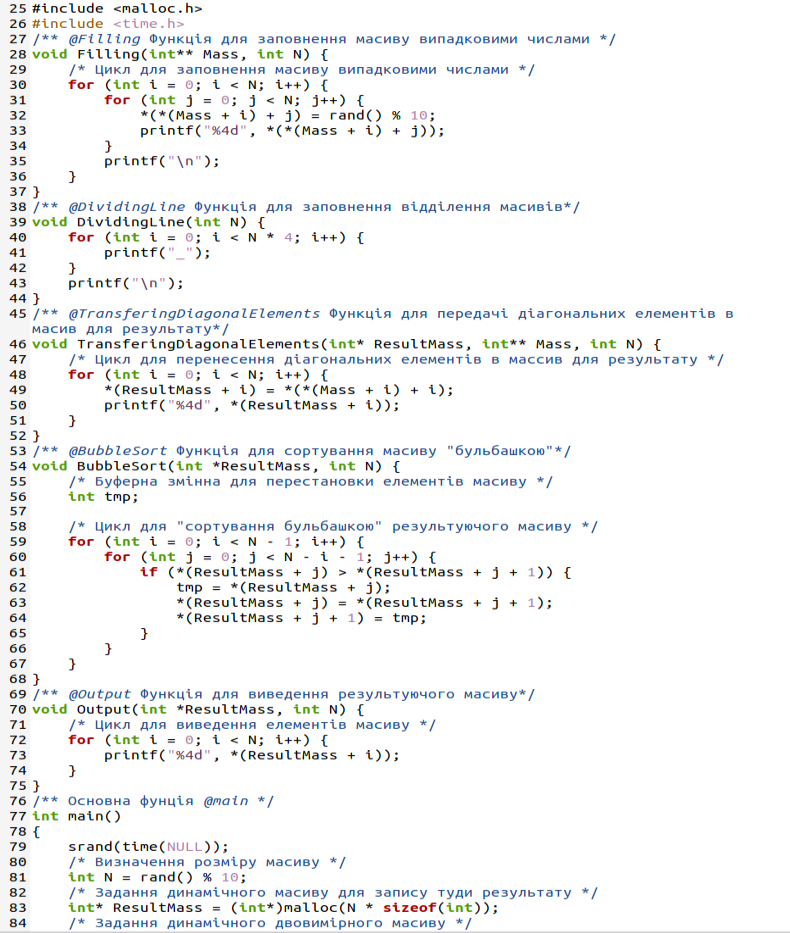




Рисунок 4 – код

**2.4** Зробив компіляцію проекту та оцінив його роботоздатність.

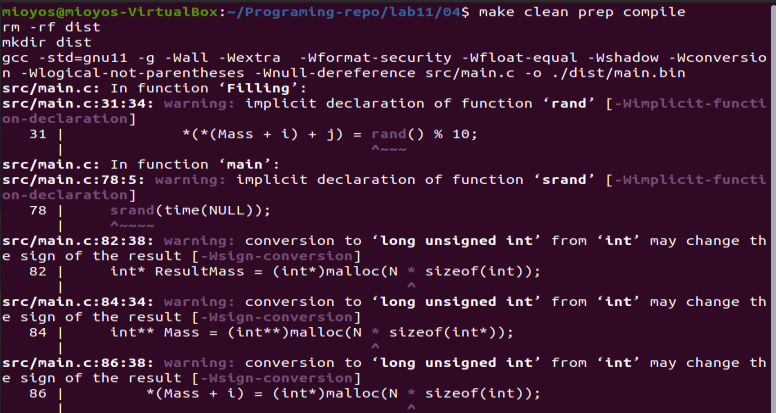


Рисунок 5 – компіляція

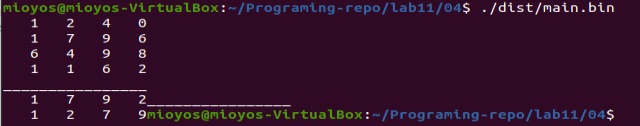
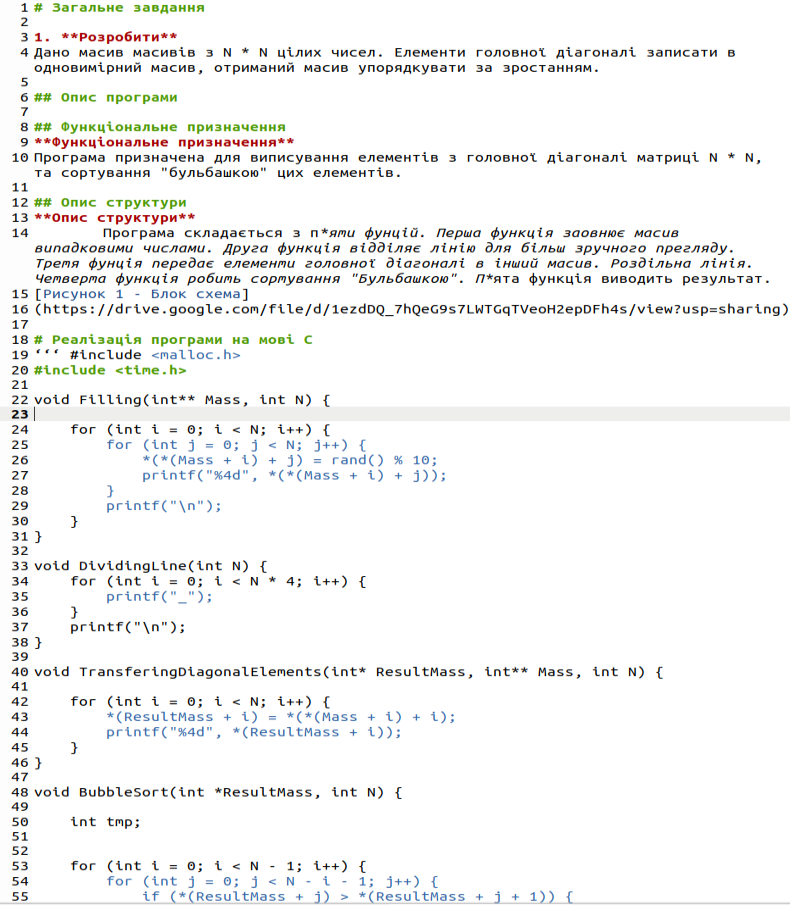


Рисунок 6 – тест

**2.5** Написав звіт до файлу в форматі MarkDown.



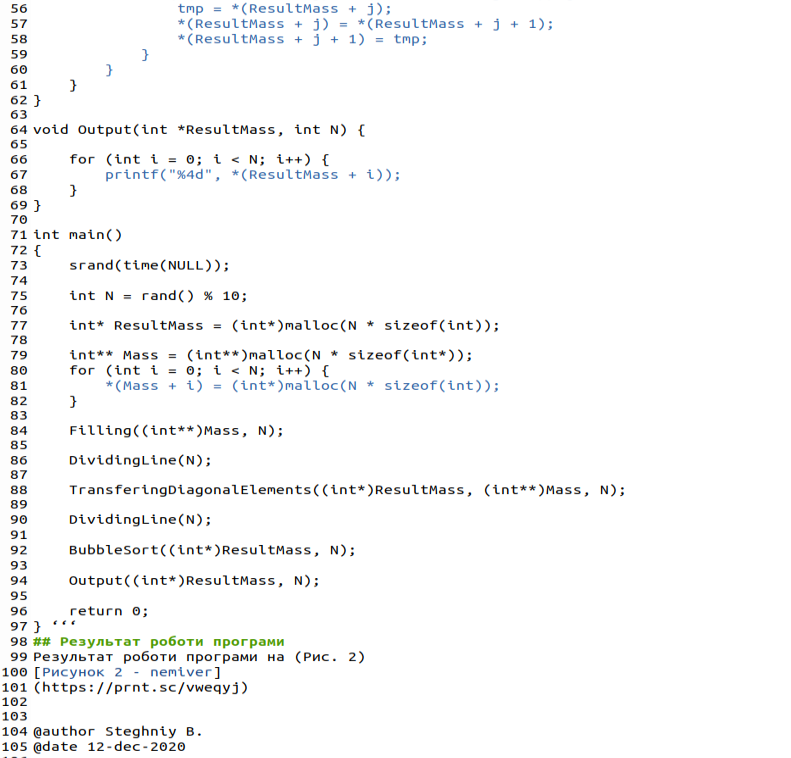


Рисунок 7 – MarkDown

**2.6** Згенерував html файл за допомогою команди make doxy



Рисунок 8 – make doxy

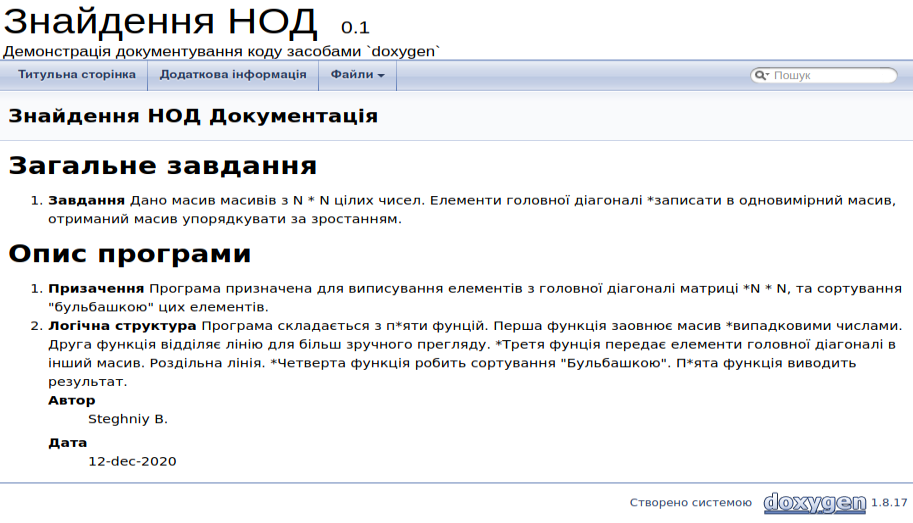


Рисунок 9 – Doxygen

**2.7** Усі зміни були додані, зафіксовані та завантажені до онлайн-репозиторію Programing-repo за допомогою команд git add, git commit, git push.



Рисунок 10 – git add

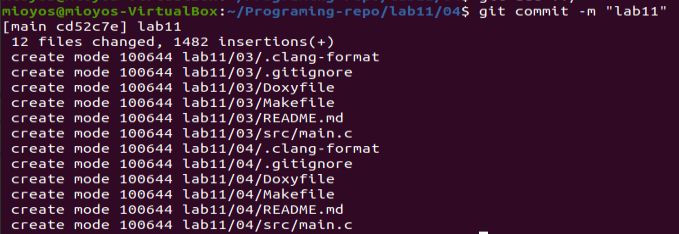


Рисунок 11 – git commit

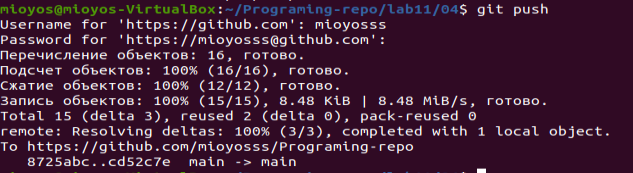


Рисунок 12 – git push

**2.8** Зробив блок-схему для програми.

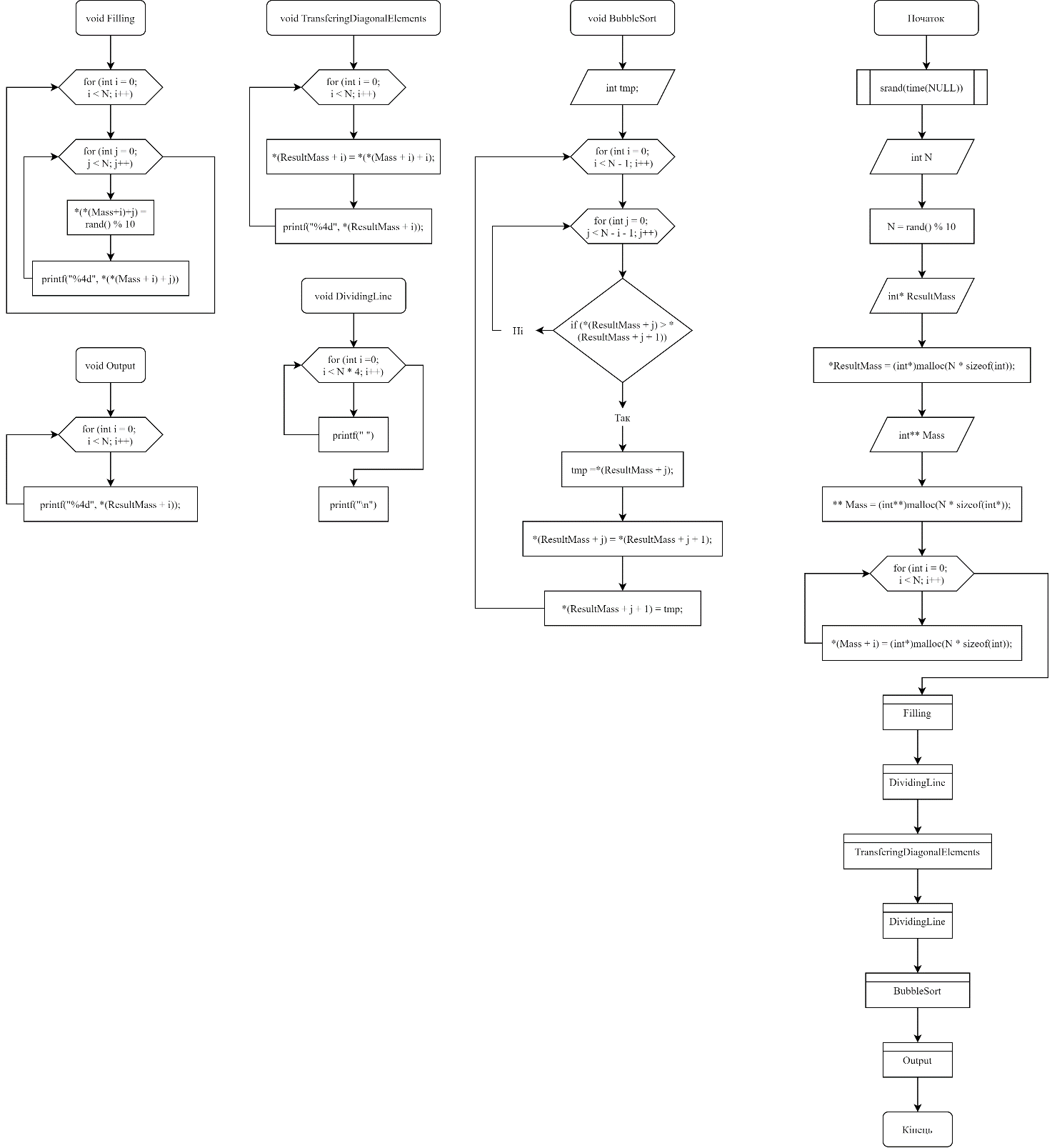
****

Рисунок 13 – блок-схема

**Висновки**

При виконанні лабораторної роботи № 11 «Вступ до показчиків», розробив дві програми таким чином, щоб у їх виконанні були задіяні показчики та функції. Були розвинуті нові навички у роботі з середовищем Linux.

**Контрольні питання**

1. За допомогою \*. Наприклад, \*(arr + 1).

2. Операція розіменування «\*»;

Операція взяття адреси «&»;

Операція присвоювання «=»;

Операції інкремент «++» і декремент « –»;

Операції додавання «+» і віднімання «-»;

Операції відношення (порівняння) покажчиків однакового типу: «==», «!= », «<», «<=», «>», «>=».

3. За допомогою функції free().

4. За допомого операції malloc(масив буде заповнений сміттям) або calloc(масив буде заповнений нулями).

5. а) Відмова від динамічної пам’яті.

б) Розумні вказівники (дозволяють дещо узгодити час життя вказівника і час життя об'єкта, на який він посилається).

в) Очищувати пам’ять своєчасно.

г) Перезапуск програми.

6. При статичному виділенні вивільнення пам’яті відбувається автоматично при закінченні роботи програми і програміст не має змоги виконати вивільнення пам’яті під час роботи програми. При динамічному виділенні вивільнення пам’яті також відбувається автоматично при закінченні роботи програми, але програміст може при необхідності вивільнити пам’ять в процесі виконання програми. Це робить програми більш гнучкими, тому в сучасному програмуванні все частіше використовують динамічне виділення пам’яті при роботі з масивами.

7. За допомогою операції sizeof.

8. 1) Присвоюємо показчику адресу нульового елементу масиву.

2) Задаємо цикл. Поки змінна і менше розміру масиву, збільшуємо покажчик.

9. Для того, щоб пам’ять не засмічувалась та не відбувався Memory leak.