

Міністерство освіти і науки України  
Національний університет «Львівська політехніка»  
Кафедра систем штучного інтелекту



## Звіт

### **про виконання лабораторних та практичних робіт блоку № 4**

На тему: «Одновимірні масиви. Двовимірні Масиви. Вказівники та Посилання.  
Динамічні масиви. Структури даних. Вкладені структури. Алгоритми обробки та  
робота з масивами та структурами.»

**з дисципліни:** «Основи програмування»

до:

ВНС Лабораторної Роботи №4

ВНС Лабораторної Роботи №5

Алготестер Лабораторної Роботи №2

Алготестер Лабораторної Роботи №3

Практичних Робіт до блоку №4

**Виконав:**

Студент групи ІІІ-13

Скічко Михайло Вікторович

## Тема роботи:

Одновимірні масиви. Двовимірні Масиви. Вказівники та Посилання. Динамічні масиви. Структури даних. Вкладені структури. Алгоритми обробки та робота з масивами та структурами.

## Мета роботи:

Навчитись використовувати масиви, вказівники та посилання, організовувати структури даних. Засвоїти на практиці алгоритми обробки та роботи з масивами та структурами ивчення символічних і рядкових змінних і способів їхньої обробки в мові С. Робота із двійковими файлами, організація вводу-виводу структурованої інформації і її зберігання на зовнішніх носіях. Робота з текстовими файлами, ввід-вивід текстової інформації і її зберігання на зовнішніх носіях.

## Теоретичні відомості:

### 1) Теми, необхідні для виконання роботи:

1. Класи пам'яті у C++
  - Статична пам'ять.
  - Динамічна пам'ять.
  - Поняття стеку.
  - Виділення та вивільнення пам'яті.
2. Вступ до Масивів і Вказівників:
  - Основи масивів: визначення, важливість, приклади використання.
  - Різниця між статичними та динамічними масивами.
  - Основи вказівників: що це таке, як вони працюють.
  - Взаємозв'язок між масивами та вказівниками.
  - Вступ до посилань: основні концепції та відмінності від вказівників.
3. Одновимірні Масиви:
  - Створення та ініціалізація одновимірних масивів.
  - Основні операції: індексація, присвоєння, читання.
  - Цикли та обхід масивів.
  - Використання функцій для роботи з масивами.
  - Приклади алгоритмів сортування та пошуку.
4. Вказівники та Посилання:
  - Використання вказівників для доступу до елементів масиву.
  - Арифметика вказівників.
  - Різниця між вказівниками та посиланнями в контексті функцій.
  - Динамічне виділення пам'яті з використанням вказівників.
  - Використання вказівників для створення складних структур даних.
5. Двовимірні Масиви:
  - Оголошення та ініціалізація двовимірних масивів.
  - Вкладені цикли для обходу двовимірних масивів.
  - Практичні приклади використання двовимірних масивів.
  - Передача двовимірних масивів у функції.
  - Застосування двовимірних масивів для розв'язання задач.

6. Динамічні Масиви:
  - Основи динамічного виділення пам'яті.
  - Створення та управління динамічними масивами.
  - Використання операторів new та delete для управління пам'яттю.
  - Реалізація змінної розмірності масивів.
  - Передача динамічних масивів у функції.
7. Структури Даних:
  - Оголошення та використання структур.
  - Використання масивів та вказівників у структурах.
  - Функції для обробки даних у структурах.
  - Використання структур для представлення складних даних.
  - Вкладені структури та їх використання.
  - Об'єднання (Union)
  - Переліки (enumerations)
8. Вкладені Структури:
  - Поняття вкладених структур та їх оголошення.
  - Взаємодія з вкладеними структурами.
  - Використання вкладених структур для моделювання складних даних.
  - Передача вкладених структур у функції.
  - Приклади реального використання вкладених структур.
9. Використання структур
  - Перевантаження операторів у структурі.
  - Вивід/ввід структури (operator<<);
  - Арифметичні операції з структурами (operator+, operator-);
  - Практичні задачі на виведення структур та операції з ними
10. Алгоритми обробки та робота з масивами та структурами:
  - Алгоритми пошуку та сортування в масивах.
  - Обробка та маніпуляції з даними у структурах.
  - Використання циклів та умовних операторів для роботи з масивами та структурами.
  - Інтеграція масивів та структур у алгоритми.
  - Розв'язання практичних задач з використанням масивів та структур.

## **2) Джерела використані для ознайомлення з вищезазначеними темами:**

- Всю інформацію до теоретичних відомостей я отримав на лекційних, практичних парах, та самостійне вивчення. Зокрема сайти <https://acode.com.ua/> та <https://www.w3schools.com/>

# Виконання роботи:

## Опрацювання завдання та вимог до програм та середовища:

### Завдання №1 – VNS Lab 4 Task 1 variant 23

#### Задача

23.

- 1) Реалізувати з використанням масиву двонаправлене кільце (перегляд можливий в обидва боки, від останнього елемента можна перейти до першого).
- 2) Роздрукувати отриманий масив, починаючи з K-ого елемента і до K-1 (по кільцю вліво).
- 3) Знищити з кільця всі елементи, що співпадають з його максимальним значенням.
- 4) Роздрукувати отриманий масив, починаючи з K-ого елемента (і до K+1 по кільцю вправо).

### Завдання №2– VNS Lab 5 Task 1 variant 23

#### Задача

Використовуючи функції, розв'язати зазначене у варіанті завдання. Масив повинен передаватися у функцію як параметр.

23. Задано одновимірний масив, який складається з N цілих чисел. Сформувати на його основі двовимірний масив  $N \times N$  так, щоб сума елементів у першому стовпці дорівнювала першому елементу одновимірного масиву, сума елементів у другому стовпці повинна дорівнювати другому елементу одновимірного масиву й т.ін. Нулі не використовувати.

## Завдання №3 – Algotester Lab 2 variant 2

### Задача

#### Lab 2v2

*Limits: 1 sec., 256 MiB*

У вас є масив  $r$  розміром  $N$ . Також вам дано 3 цілих числа.

Спочатку ви маєте видалити з масиву ці 3 числа, які вам дані. Після цього перетворити цей масив у масив сум, розміром  $N_{new} - 1$  (розмір нового масиву після видалення елементів), який буде відображати суми сусідніх елементів нового масиву.

Далі необхідно вивести масив сум на екран.

#### Input

У першому рядку ціле число  $N$  - кількість чисел

У другому рядку масив  $r$ , який складається з  $N$  цілих чисел

У третьому рядку 3 цілих числа,  $a, b, c$ , які треба видалити з масиву

#### Output

У першому рядку ціле число  $M$  - кількість чисел у масиві, який буде виведено

У наступному рядку  $M$  чисел - новий масив

#### Constraints

$$1 \leq N \leq 10^5$$

$$0 \leq r_i \leq 10^5$$

$$0 \leq a, b, c \leq 10^5$$

## Завдання №4 – Algotester Lab 3 variant 2

### Задача

#### Lab 3v2

*Limits: 1 sec., 256 MiB*

Вам дано 2 масиви розміром  $N$  та  $M$ . Значення у цих масивах унікальні.

Ваше завдання вивести у першому рядку кількість елементів, які наявні в обох масивах одночасно, у другому кількість унікальних елементів в обох масивах разом.

#### Input

У першому рядку ціле число  $N$

у другому рядку  $N$  цілих чисел  $a_1 \dots a_n$

У третьому рядку ціле число  $M$

у четвертому рядку  $M$  цілих чисел  $b_1 \dots b_n$

#### Output

У першому рядку одне ціле число - кількість елементів, які наявні в обох масивах одночасно.

У другому рядку кількість унікальних елементів в обох масивах (тобто кількість унікальних елементів у масиві, який буде об'єднанням двох даних).

#### Constraints

$$0 \leq N \leq 100$$

$$0 \leq a_i \leq 100$$

$$0 \leq M \leq 100$$

$$0 \leq b_i \leq 100$$

## Завдання №5 – Class Practice Work

### Задача

# Перевірка чи слово або число є паліндромом

### Задача

Реалізувати програму, яка перевіряє, чи дане слово чи число є паліндромом за допомогою рекурсії.

Паліндром — це послідовність символів, яка однаково читається вперед і назад (наприклад, «radar», «level», «12321»).

### Мета Задачі

Навчитися користуватися механізмами перевантаження функції та використовувати рекурсію для вирішення задач обчислення.

### Вимоги:

1. Визначення функції:
  - a. Реалізуйте рекурсивну функцію *isPalindrome*, яка перевіряє, чи заданий рядок є паліндромом.
2. Приклад визначення функції:
  - a. `bool isPalindrome(const string& str, int start, int end);`
3. Перевантаження функцій:
  - a. Перевантажте функцію *isPalindrome* для роботи з цілими значеннями.
  - b. `bool isPalindrome(ціле число);`
4. Рекурсія:
  - a. Рекурсивна функція для рядків перевірить символи в поточній початковій і кінцевій позиціях. Якщо вони збігаються, він буде рекурсивно перевіряти наступні позиції, поки початок не перевищить кінець, після чого рядок буде визначено як паліндром.

### Кроки реалізації

- Визначте та реалізуйте рекурсивну функцію *isPalindrome* для рядків.
- Визначте та реалізуйте перевантажену функцію *isPalindrome* для цілих чисел. Використати математичний підхід щоб перевірити чи число є паліндромом.

Дизайн та планована оцінка часу виконання завдань:

**Завдання №1 – VNS Lab 4 Task 1 variant 23**

Планований час виконання: 30 хв

**Завдання №2– VNS Lab 5 Task 1 variant 23**

Планований час виконання: 50 хв

**Завдання №3 – Algotester Lab 2 variant 2**

Планований час виконання: 20 хв

**Завдання №4 – Algotester Lab 3 variant 2**

Планований час виконання: 40 хв

**Завдання №5 – Class Practice Work**

Планований час виконання: 50 хв

Код програм з посиланням на зовнішні ресурси:

**Завдання №1 – VNS Lab 4 Task 1 variant 23**

файл vns\_lab\_4\_task\_1\_variant\_23\_mykhailo\_skichko.cpp

**Завдання №2– VNS Lab 5 Task 1 variant 23**

файл vns\_lab\_5\_task\_1\_variant\_23\_mykhailo\_skichko.cpp

**Завдання №3 – Algotester Lab 2 variant 2**

файл algotester\_lab\_2\_variant\_2\_mykhailo\_skichko.cpp

**Завдання №4 – Algotester Lab 3 variant 2**

файл algotester\_lab\_3\_variant\_2\_mykhailo\_skichko.cpp

**Завдання №5 – Class Practice Work**

файл practice\_work\_team\_tasks\_mykhailo\_skichko.cpp



Результати виконання завдань, тестування  
та фактично затрачених час:

**Завдання №1 – VNS Lab 4 Task 1 variant 23**

Фактично затрачених час: 50 хв

```
Original Circular Array: Circular Array: 5 3 8 6 8 2
Task 2: Starting from Kth element to the left
Circular Array: 8 3 5 2 8 6
Task 3: Removing elements matching the maximum value
Task 4: Starting from Kth element to the right
Circular Array: 6 2 5 3
```

**Завдання №2 – VNS Lab 5 Task 1 variant 23**

Фактично затрачених час: 40 хв

```
Resulting 2D array:
1 1 1
1 1 1
4 8 13
```

**Завдання №3 – Algotester Lab 2 variant 2**

Фактично затрачених час: 30 хв

Created	Compiler	Result	Time (sec.)	Memory (MiB)	Actions
a day ago	C++ 23	Accepted	0.003	1.410	<a href="#">View</a>

**Завдання №4 – Algotester Lab 3 variant 2**

Фактично затрачених час: 40 хв

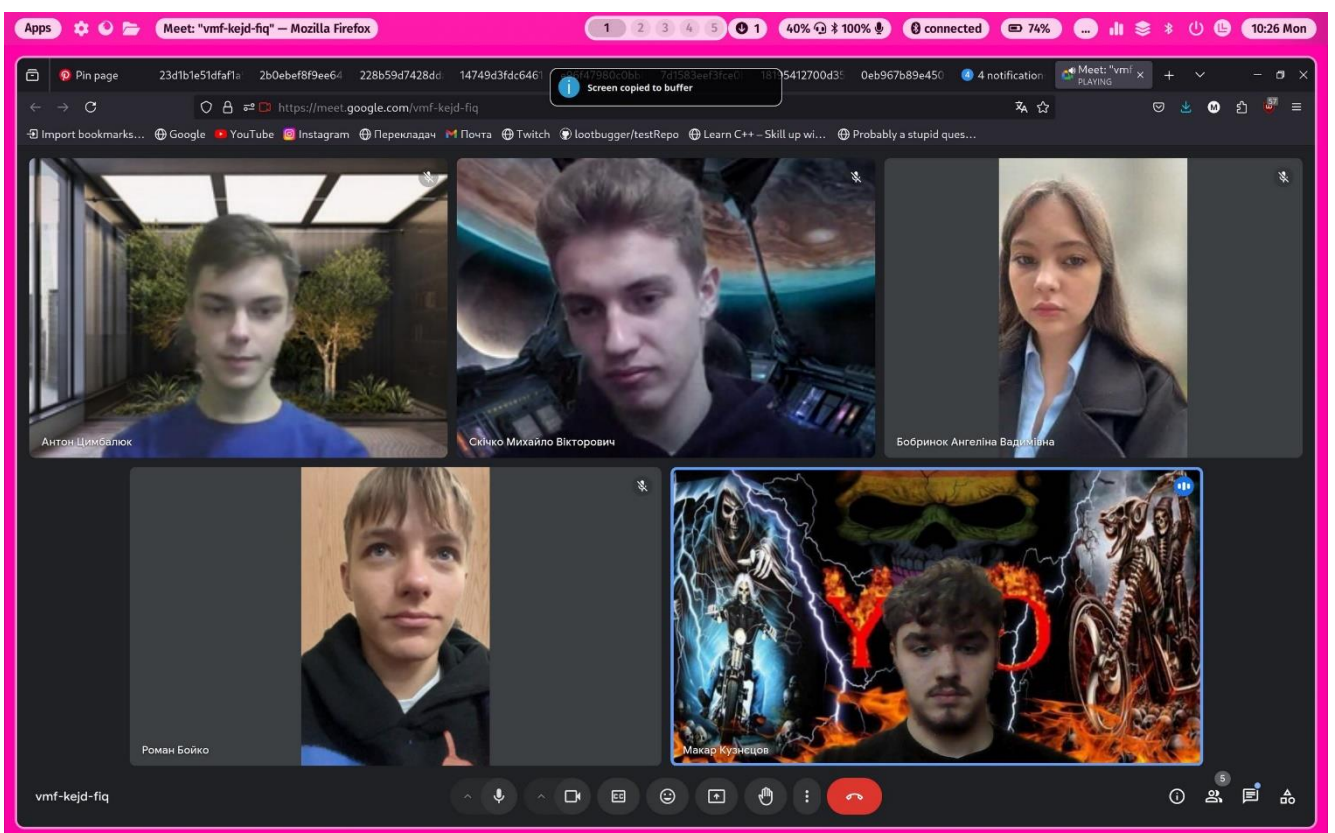
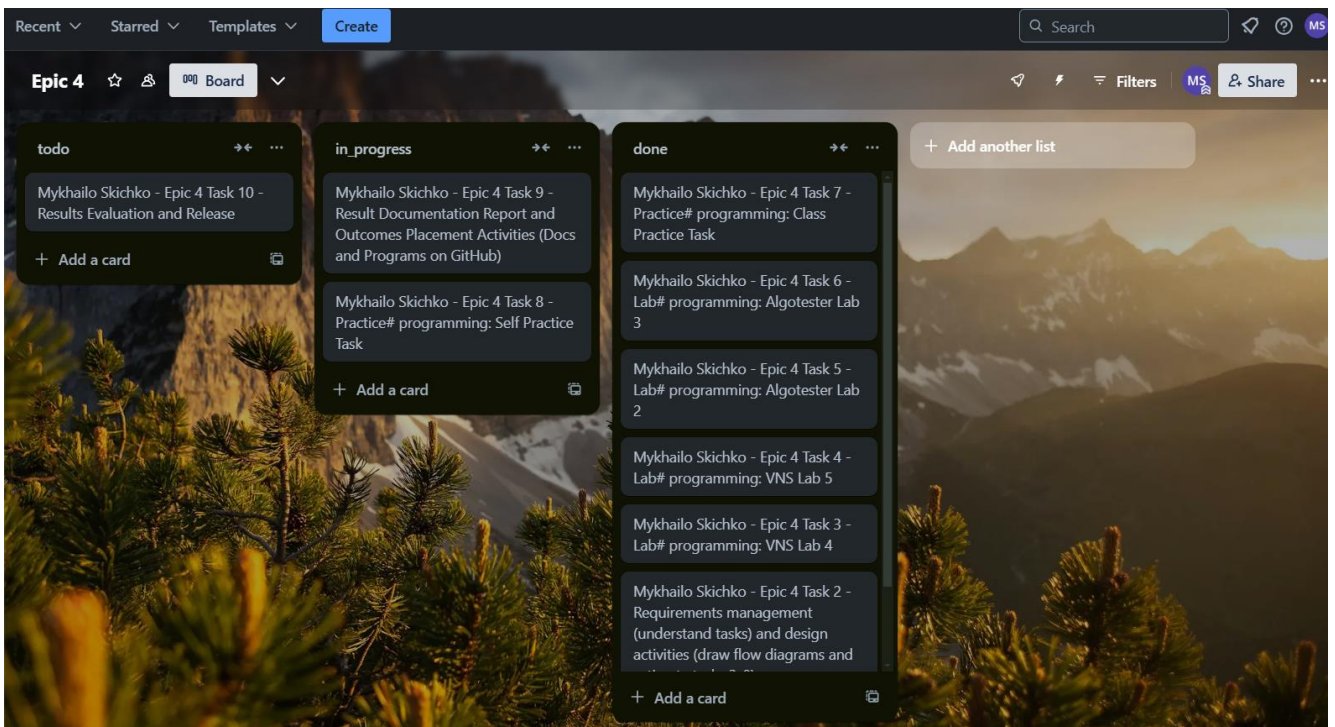
Created	Compiler	Result	Time (sec.)	Memory (MiB)	Actions
a day ago	C++ 23	Accepted	0.003	1.195	<a href="#">View</a>

**Завдання №5 – Class Practice Work**

Фактично затрачених час: 70 хв

```
Enter a string to check if it's a palindrome: avadodava
The string "avadodava" is a palindrome.
Enter a number to check if it's a palindrome: 7439347
The number 7439347 is a palindrome.
```

## Кооперація з командою:



## Висновки:

Під час виконання цієї роботи я засвоїв основні принципи роботи з одновимірними та двовимірними масивами, вказівниками, посиланнями, динамічними масивами, а також зі структурами даних та вкладеними структурами. Я навчився використовувати масиви та вказівники для ефективного доступу до пам'яті, а також зрозумів, як статичне і динамічне виділення пам'яті допомагають управляти ресурсами програми.

Окрім того, мені вдалося зрозуміти важливість структур для організації та обробки складних даних. Практика з алгоритмами сортування та пошуку для масивів і структур показала, як вони можуть пришвидшити обробку даних і спростити написання коду. Завдяки виконанню практичних завдань я отримав глибше розуміння того, як інтегрувати масиви та структури у більш складні алгоритми та вирішувати реальні завдання.

Посилання на pull request: