Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра систем штучного інтелекту

A blue and white logo

Description automatically generated

**Звіт**

**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

**Виконала:**

Студентка групи ШІ-13

Козак Наталія Володимирівна

Львів 2024

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

Теоретичні відомості:

1. Класи пам'яті у C++

-        Статична пам’ять.

-        Динамічна пам’ять.

-        Поняття стеку.

-        Виділення та вивільнення пам’яті.

1. 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.

1. 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.

1. 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.

1. 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.

1. 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.

1. 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)

1. 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

1. 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.

Виконання роботи:

1. Опрацювання завдання та вимог до програм:

**Завдання №1 VNS Lab 4 - Task 1-19**

1) Реалізувати з використанням масиву однонаправлене кільце (перегляд

можливий зправа наліво, від першого елемента можна перейти до

останнього).

2) Роздрукувати отриманий масив, починаючи з К-ого елемента і до К+1.

3) Додати в кільце перший і останній елементи.

4) Знищити з кільця непарні елементи.

5) Роздрукувати отриманий масив, починаючи з К-ого елемента й до К+1.

**Завдання №2 VNS Lab 5 - Task 1-19**

Задано рядок з N^2 цифр. Встановити чи можна, розбивши рядок на

підстрічки довжиною N, записати їх у рядки двовимірного масиву N x N по

одній цифрі в одному елементі так, щоб вони в першому стовпці

розташувалися в порядку зростання.

**Завдання №3 Algotester Lab 2v3**

Вам дано масив цілих чисел розміром N, на першій та останній клітинці розміщено по дрону.

Вони одночасно взлітають.

На початку кожного ходу швидкість дрону стає рівною значенню клітинки, у якій він знаходиться.

Тобто лівий дрон у першу секунду з клітинки з індексом 1 перелетить у клітинку з індексом a1, тобто його наступна позиція рахується як поточна позиція + число у поточній позиції (перегляньте пояснення для візуалізації) Правий робить аналогічно в протилежну сторону.

Вони це роблять до моменту, коли трапиться одна з зазначених подій:

Якшо 2 дрони опиняються в одній клітинці - ви виводите **Collision**.

Якщо лівий дрон опиниться справа від правого - це **Miss**

У випадку якщо вони зупиняться один навпроти одного, тобто у клітинках ai та ai+1 - виведіть **Stopped**

Врахуйте, що перевіряти треба також до взльоту.

**Завдання №4 Algotester Lab 3v2**

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.

**Завдання №5 Class Practice Work**

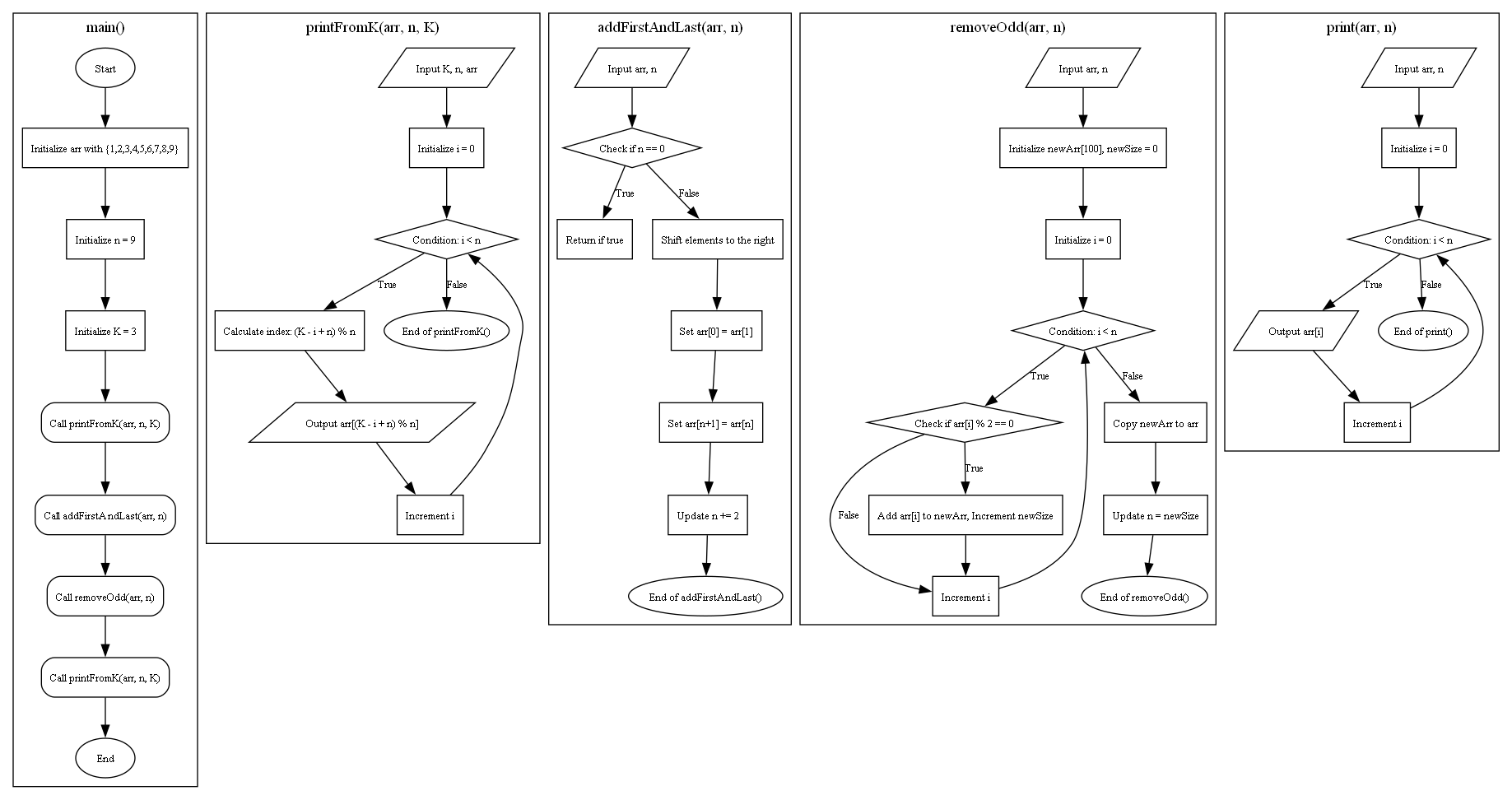
Реалізувати програму, яка перевіряє, чи дане слово чи число є паліндромом за

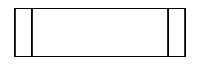
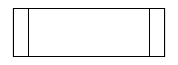
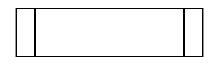
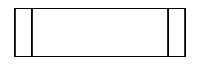
допомогою рекурсії.

**Завдання №6 Self Practice Work**

Задано рядок s. Дозволено переставляти літери рядка s між собою.

Чи можна перетворити рядок s на паліндром?

****2. Дизайн та планована оцінка часу виконання завдання №1:



Планований час на реалізацію : 20хв

3. Код програм і фактично затрачений час

**Завдання №1 VNS Lab 4 - Task 1-19**

*ai\_13/nataliia\_kozak/epic\_4/code/vns\_lab\_4\_task\_1\_variant\_19\_natalya\_kozak.cpp*

Фактично затрачений час – 30хв

**Завдання №2 VNS Lab 5 - Task 1-19**

*ai\_13/nataliia\_kozak/epic\_4/code/vns\_lab\_5\_task\_1\_variant\_19\_natalya\_kozak.cpp*

Фактично затрачений час – 20хв

**Завдання №3 Algotester Lab 2v3**

*ai\_13/nataliia\_kozak/epic\_4/code/algotester\_lab\_2\_task\_3\_natalya\_kozak.cpp*

Фактично затрачений час – 25хв

**Завдання №4 Algotester Lab 3v2**

*ai\_13/nataliia\_kozak/epic\_4/code/algotester\_lab\_3\_task\_2\_natalya\_kozak.cpp*

Фактично затрачений час – 35хв

**Завдання №5 Class Practice Work**

*ai\_13/nataliia\_kozak/epic\_4/code/practice\_work\_task\_1\_natalya\_kozak.cpp*

Фактично затрачений час – 30хв

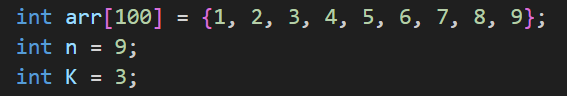
**Завдання №6 Self Practice Work**

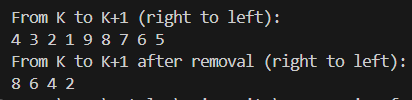
*ai\_13/nataliia\_kozak/epic\_4/code/practice\_work\_self\_algotester\_tasks\_natalya\_kozak.cpp*

Фактично затрачений час – 15хв

4. Результати виконання завдань і тестування:

**Завдання №1 VNS Lab 4 - Task 1-19**

****

****

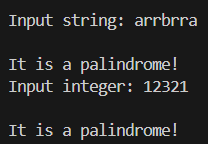
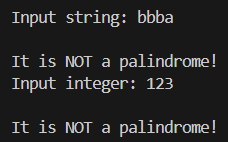
**Завдання №2 VNS Lab 5 - Task 1-19**

**** ****

**Завдання №3 Algotester Lab 2v3**

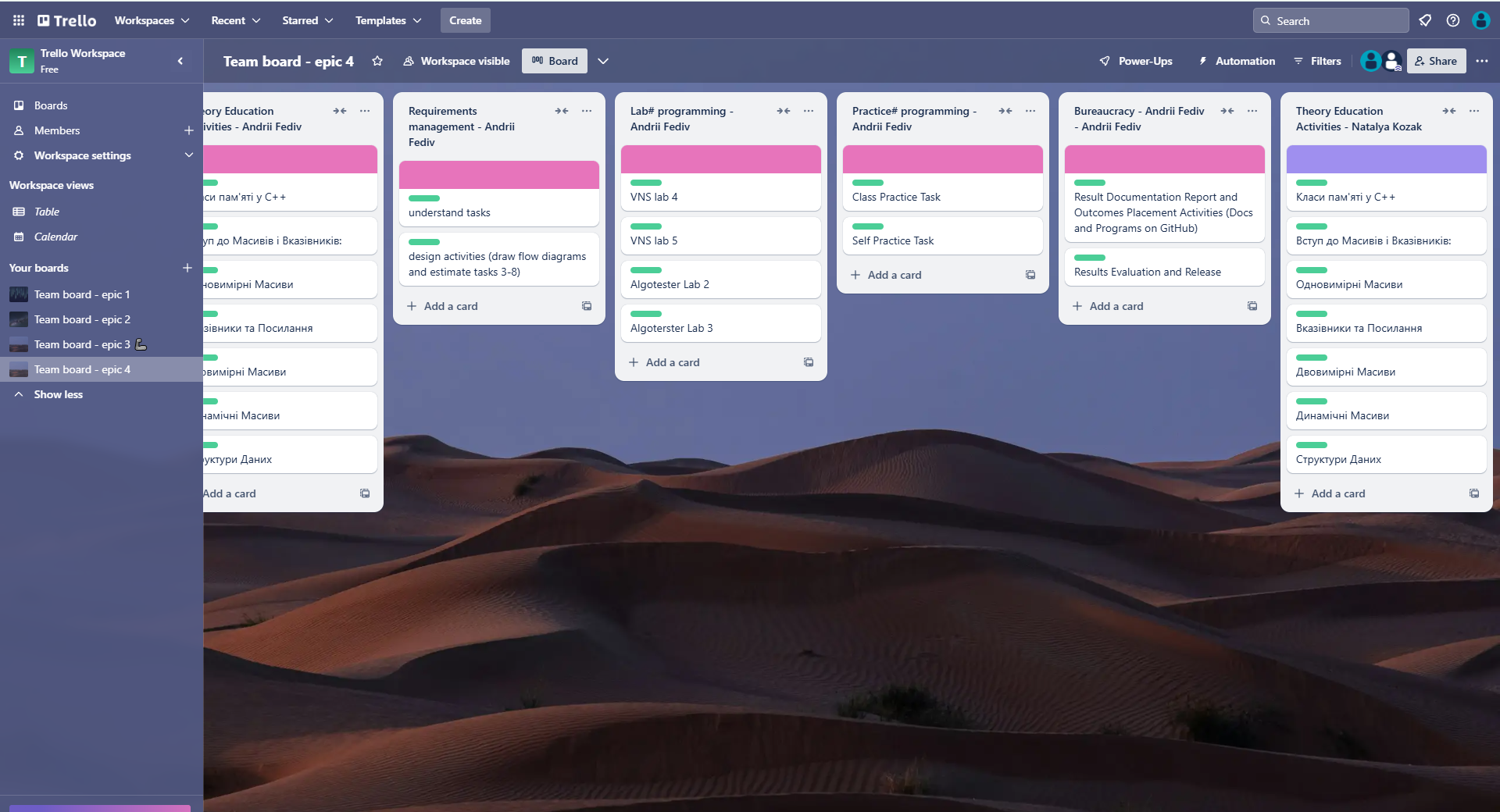
**Завдання №4 Algotester Lab 3v2**

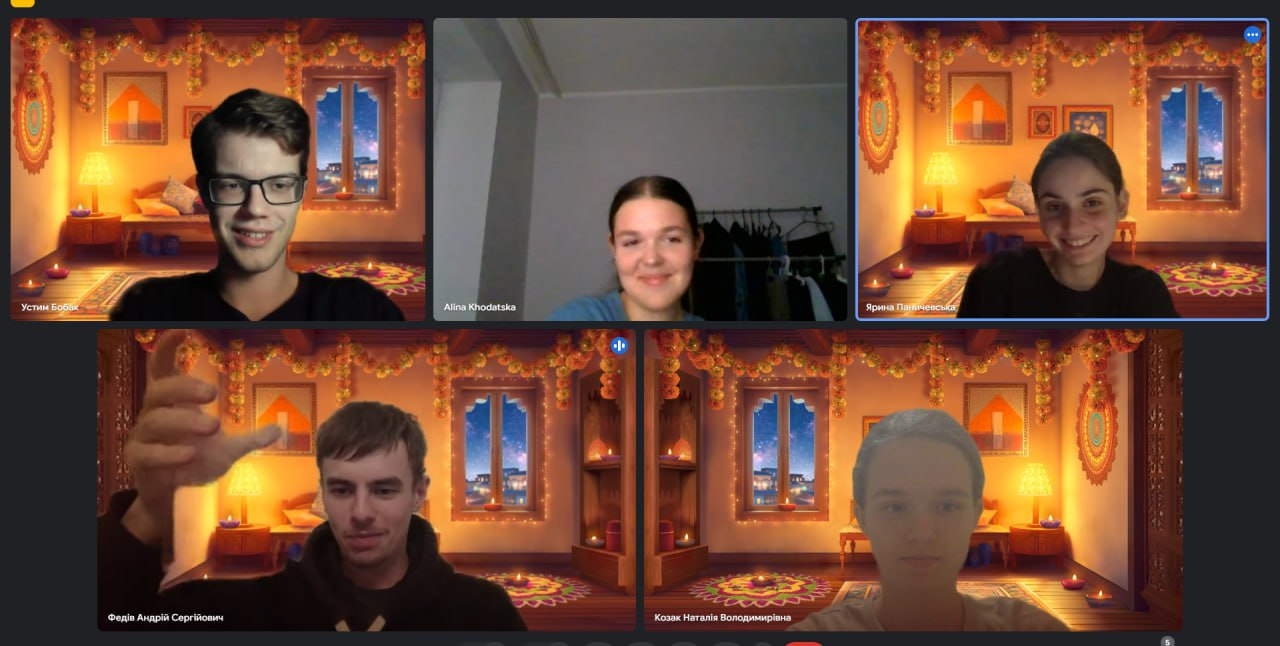
**Завдання №5 Class Practice Work**

**** ****

**Завдання №6 Self Practice Work**

5. Кооперація з командою





Висновки:

Під час виконання цієї роботи я засвоїла основні принципи роботи з одновимірними та двовимірними масивами, вказівниками, посиланнями, динамічними масивами, а також зі структурами даних та вкладеними структурами. Я навчилася використовувати масиви та вказівники для ефективного доступу до пам’яті, а також зрозуміла, 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 отримала глибше розуміння того, як інтегрувати масиви та структури у більш складні алгоритми та вирішувати реальні завдання.