**Міністерство освіти і науки України**

N

1

a1

ai

ai+1

N

N

N

N

a1..an

M

M

b1..bn

N

N

a1..an

M

M

b1..bn

a

n

b

m

k=min(n,m)

c

a

d

b

∑i=1k|ci−di|.

n

m

a

b

n

ai

m

bi

**Національному університеті**

**"Львівська Політехніка"**

**Кафедра систем штучного інтелекту**

Звіт з лабораторної роботи №4

з дисципліни

≪ Основи програмування ≫

**Виконав:**

студент групи ШІ-13

Цимбалюк Антон

**Викладач:**

Пшеничний О.Ю

Львів — 2024р.

**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:**

Навчитися працювати з одновимірними, двовимірними та динамічними масивами, вказівниками та посиланнями, структурами та вкладеними структурами.

**Теоретичні відомості:**

1. **Теми, необхідні для виконання роботи:**

* Одновимірні та двовимірні масиви
* Динамічні масиви
* Структури та вкладені структури
* Вказівники та посилання

1. **Джерела використані для ознайомлення з вищезазначеними темами:**

* З усією необхідною інформацією я був знайомий з власного досвіду.

**Виконання роботи:**

1. *Опрацювання завдання та вимог до програми та середовища*

**Завдання №1 - Class Practice Work – Перевірка чи слово або число є паліндромом**

**Задача**

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

## Мета Задачі

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

### Вимоги:

1. Визначення функції:

Реалізуйте рекурсивну функцію *isPalindrome*, яка перевіряє, чи заданий рядок є паліндромом.

1. Приклад визначення функції:

*bool isPalindrome(const string& str, int start, int end);*

1. Перевантаження функцій:

Перевантажте функцію *isPalindrome* для роботи з цілими значеннями.

*bool isPalindrome*(ціле число);

1. Рекурсія:

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.

**Завдання №2****– VNS Lab 4 - Task 1-25**

**Задача:**

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) Впорядкувати елементи по спаданню.

4) Знищити з кільця непарні елементи.

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

**Завдання №3 - VNS Lab 5 - Task 1-25**

**Задача:**

Із двовимірного масиву в одновимірний записали спочатку рядки в довільному порядку, потім стовпці в довільному порядку. Написати програму, яка відновлює вихідний двовимірний масив маючи одновимірний, якщо відомо розмірність двовимірного масиву й елементи в ньому не повторюються.

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

**Задача:**

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  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.

# Input

У першому рядку ціле число N - розмір масиву

У другому рядку N цілих чисел - елементи масиву

# Output

У першому рядку фінальна позиція першого та другого дрона.

У другому рядку одне зі слів

**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 a1..an

У третьому рядку ціле число M

у четвертому рядку M цілих чисел b1..bn

# Output

У першому рялку 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).

**Завдання №5 – Algotester lab 3 variant 2**

**Задача:**

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 a1..an

У третьому рядку ціле число M

у четвертому рядку M цілих чисел b1..bn

# Output

У першому рялку 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).

**Завдання №5 — Algotester self practice work** permitations of two array

# Permutations of Two Arrays

*Limits: 2 sec., 512 MiB*

You are given two sequences: a of length n and b of length m.

Let k=min(n,m).

You want to choose a permutation c of the sequence a and a permutation d of the sequence b to maximize the following score:k∑i=1|ci−di|.

# Input

The first line contains two integers n and m – the lengths of the sequences a and b, respectively.

The second line contains n integers ai.

The third line contains m integers bi.

# Output

In the single line print an integer – the maximum score.

*2) Дизайн та планована оцінка часу виконання завдань*

**Завдання №1 - Class Practice Work - Менеджмент бібліотеки**

Дизайн алгоритму завдання знаходиться в папці “ Diagrams ” під назвою practice\_work\_task\_1\_vladyslav\_murashko.drawio

Попередньо розраховував, що це завдання займе 30 хвилин

**Завдання №2 - VNS Lab 4 - Task 1-25**

Попередньо розраховував, що це завдання займе 1 годину

**Завдання №3 - VNS Lab 5 - Task 1-25**

Попередньо розраховував, що це завдання займе 1 годину

**Завдання №4 - Algotester lab 2 variant 3**

Попередньо розраховував, що це завдання займе 30 хв

**Завдання №5 – Algotester lab 3 variant 2**

Попередньо розраховував, що це завдання займе 30 хв

**Завдання №6 – Self practice work**

Попередньо розраховував, що це завдання займе 30 хв

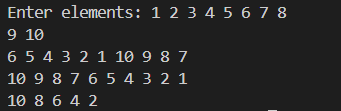
*Результати виконаних завдань, тестування та фактично затрачений час*

**Завдання №1 - Class Practice Work**



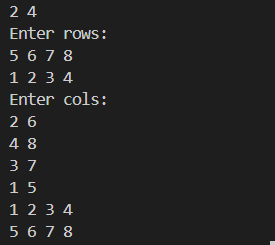
Зайняло часу – 20 хвилин

**Завдання №2 – VNS Lab 4 - Task 1-25**



Зайняло часу – 30 хвилин

**Завдання №3 – VNS Lab 5 - Task 1-25**



Зайняло часу – 30 хвилин

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

Зайняло часу – 30 хвилин

**Завдання №5 – Algotester lab 3 variant 2**

Зайняло часу – 30 хвилин

**Завдання №6 — Permutations of two arrays**

**Зайняло часу 20 хв.**

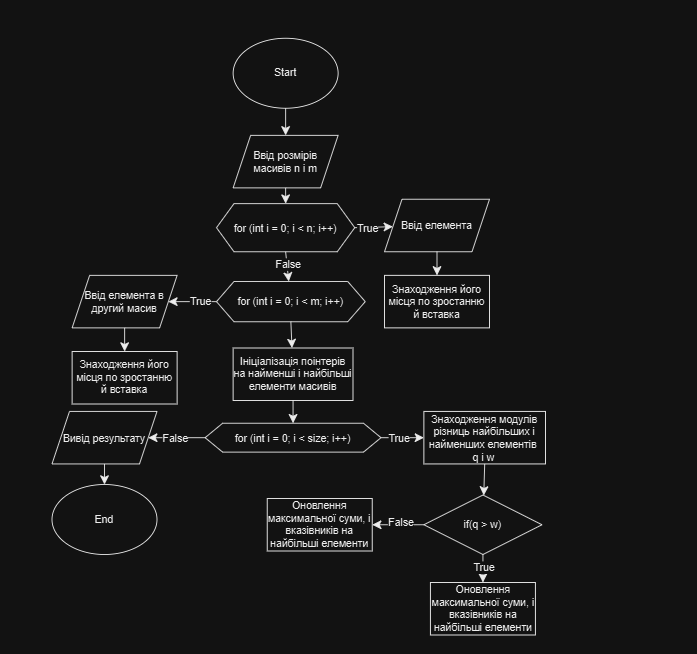
Скріншоти виконаних завдань з алготестеру



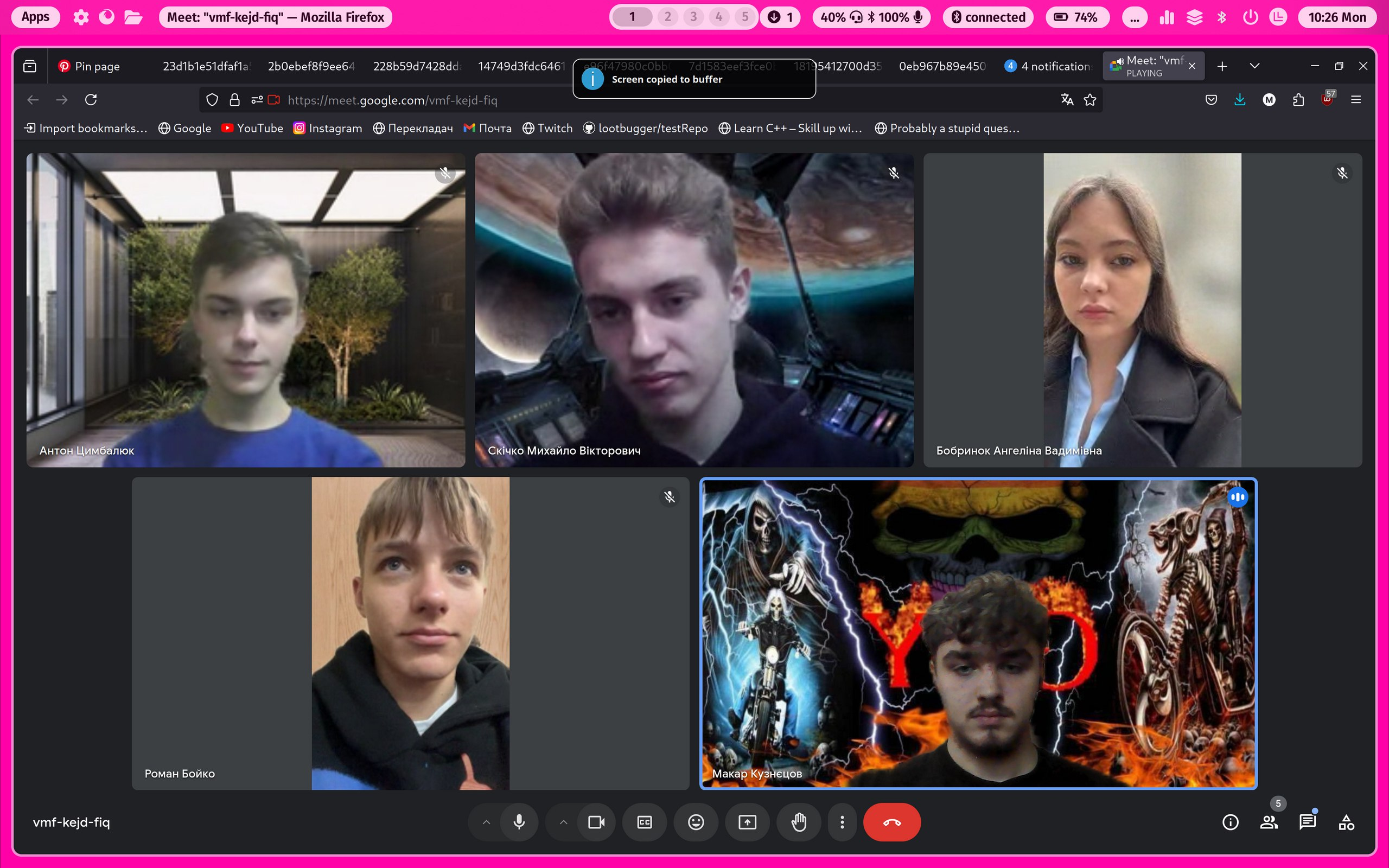


Діаграми до програм

Діаграма до завдання з самоопрацювання(20 хв):



Meet

****

**Висновки**

Під час виконання лабораторний поглибив знання С++.