Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра систем штучного інтелекту

A blue and white logo

Description automatically generated

**Звіт**

**про виконання лабораторних та практичних робіт блоку № 5**

На тему:  «Файли. Бінарні Файли. Символи і Рядкові Змінні та Текстові Файли. Стандартна бібліотека та деталі/методи роботи з файлами. Створення й використання бібліотек.»

***з дисципліни:*** «Основи програмування»

до:

ВНС Лабораторної Роботи № 6

ВНС Лабораторної Роботи № 8

ВНС Лабораторної Роботи № 9

Алготестер Лабораторної Роботи №4

Алготестер Лабораторної Роботи №6

Практичних Робіт до блоку №5

**Виконала:**

Студентка групи ШІ-13

Кшик Олена Андріївна

Львів – 2024

**Тема:** Файли. Бінарні Файли. Символи і Рядкові Змінні та Текстові Файли. Стандартна бібліотека та деталі/методи роботи з файлами. Створення й використання бібліотек.

**Мета:** Навчитись працювати з файлами, вносити у нього зміни. Опрацювати деталі роботи з файлами, набір команд у бібліотеці.

**Теоретичні відомості:**

1. Вступ до Роботи з Файлами:
   * Основні операції з файлами: відкриття, читання, запис, закриття
   * Робота з файловими дескрипторами
   * C-style читання з файлу та запис до файлу
   * Перевірка стану файлу: перевірка помилок, кінець файлу
   * Базові приклади читання та запису в файл
2. Символи і Рядкові Змінні:
   * Робота з char та string: основні операції і методи
   * Стрічкові літерали та екранування символів
   * Конкатенація, порівняння та пошук у рядках
3. Текстові Файли:
   * Особливості читання та запису текстових файлів
   * Обробка рядків з файлу: getline, ignore, peek
   * Форматування тексту при записі: setw, setfill, setprecision
   * Парсинг текстових файлів: розділення на слова, аналіз структури
   * Обробка помилок при роботі з файлами
4. Бінарні Файли:
   * Вступ до бінарних файлів: відмінності від текстових, приклади (великі дані, ігрові ресурси, зображення)
   * Читання та запис бінарних даних
   * Робота з позиціонуванням у файлі: seekg, seekp
   * Серіалізація об'єктів у бінарний формат
5. Стандартна бібліотека та робота з файлами:
   * Огляд стандартної бібліотеки для роботи з файлами
   * Потоки вводу/виводу: ifstream, ofstream, fstream
   * Обробка помилок при роботі з файлами
6. Створення й використання бібліотек:
   * Вступ до створення власних бібліотек у С++
   * Правила розбиття коду на header-и(.h) та source(.cpp) файли
   * Статичні проти динамічних бібліотек: переваги та використання
   * Інтерфейси бібліотек: створення, документування, версіонування
   * Використання сторонніх бібліотек у проектах

**Індивідуальний план опрацювання теорії:**

* [**string**](https://www.youtube.com/watch?v=1FkTJYm-T34&ab_channel=%D0%91%D0%BB%D0%BE%D0%B3%D0%B0%D0%BD)
* [**Files**](https://www.youtube.com/watch?v=FeNqHytI0fA&list=PLiPRE8VmJzOpn6PzYf0higmCEyGzo2A5g&index=166&ab_channel=%D0%91%D0%BB%D0%BE%D0%B3%D0%B0%D0%BD)
* [**Files c++**](https://www.youtube.com/watch?v=SSNJ7alki-E&list=PLiPRE8VmJzOpn6PzYf0higmCEyGzo2A5g&index=171&ab_channel=%D0%91%D0%BB%D0%BE%D0%B3%D0%B0%D0%BD)
* [**w3school**](https://www.w3schools.com/cpp/default.asp)
* **Лекції і практичні заняття**

**Виконання роботи:**

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

**VNS Lab 6 Task 1 (23)**

Задано рядок, що складається із символів. Символи поєднуються в слова.

Слова одне від одного відокремлюються одним або декількома пробілами.

Наприкінці тексту ставиться крапка. Текст містить не більше 255 символів.

Виконати ввід рядка, використовуючи функцію gets(s) і здійснити обробку

рядка у відповідності зі своїм варіантом.

23. Для рядка знайти довжину найкоротшого слова.

**VNS Lab 8 Task 1 (23)**

Сформувати двійковий файл із елементів, заданої у варіанті структури,

роздрукувати його вмістиме, виконати знищення й додавання елементів у

відповідності зі своїм варіантом, використовуючи для пошуку елементів що

знищуються чи додаються, функцію. Формування, друк, додавання й знищення

елементів оформити у вигляді функцій. Передбачити повідомлення про

помилки при відкритті файлу й виконанні операцій вводу/виводу.

23. Структура "Стадіон":

- назва;

- рік будівлі;

- кількість площадок;

- види спорту.

Знищити всі елементи, у яких рік будівлі меншее заданого, додати 2 елементи

перед елементом із зазначеним номером.

**VNS Lab 9 Task 1 (23)**

Створити текстовий файл F1 не менше, ніж з 10 рядків і записати в нього

інформацію

Виконати завдання 23

1) Скопіювати з файлу F1 у файл F2 всі рядки, у яких міститься два однакових

слова.

2) Визначити номер слова, у якому найбільше букв «А».

**Algotester Lab 4 Variant 1**

Вам дано 2 цілих чисел масиви, розміром NN та MM.

Ваше завдання вивести:

1. Різницю N-M

2. Різницю M-N

3. Їх перетин

4. Їх обєднання

5. Їх симетричну різницю

**Вхідні дані**

У першому рядку ціле число N - розмір масиву 1

У другому рядку N цілих чисел - елементи масиву 1

У третьому рядку ціле число M - розмір масиву 2

У четвертом рядку M цілих чисел - елементи масиву 2

**Вихідні дані**

Вивести результат виконання 5 вищезазначених операцій у форматі:

У першому рядку ціле число N - розмір множини

У наступному рядку N цілих чисел - посортована у порядку зростання множина

Пам’ятайте, ви маєте написати 2 варіанти розвязку, один з використанням засобів STL (std::set\_intersection, std::set\_symmetric\_difference, std::set\_difference, std::set\_union), інший зі своєю реалізацією. Своє сортування можна не писати.

**Algotester Lab 6 Variant 2**

У вас є шахова дошка розміром 8×8 та дуже багато фігур.

Кожна клітинка може мати таке значення:

* Пуста клітинка O
* Пішак P
* Тура R
* Кінь N
* Слон B
* Король K
* Королева Q

Вам дають позиції фігур на дошці (всі фігури одного кольору, кількість королів може бути > 1).

Далі йдуть Q запитів з координатами клітинки {x,y}. На кожен запит ви маєте вивести стрічку si - посортовані за алфавітом букви фігур, які атакують цю клітинку (пішаки атакують вниз).

У випадку, якщо на клітинці стоїть якась фігура - виведіть символ X.

У випадку, якщо клітинку не атакують - виведіть O.

Наявніть фігури у певній клітинці не блокує атаку для іншої фігури. Тобто якщо між турою та клітинкою стоїть інша фігура - вважається що тура атакує цю клітинку.

**Вхідні дані**

У перших 8 рядках стрічка rowi - стан i-го рядка дошки.

У наступному рядку ціле число Q - кількість записів

У наступних Q рядках 2 цілих числа x та y - координати клітинки

**Вихідні дані**

Q разів відповідь у наступному форматі:

Строка result - усі фігури, які атакують клітинку з запиту.

## **Class Practice Work**

## **Задача №1 – Запис текстової стрічки у файл із заданим ім’ям**

***Реалізувати функцію створення файла і запису в нього даних:***

enum FileOpResult { Success, Failure, … };

FileOpResult write\_to\_file(char \*name, char \*content);

*Умови задачі:*

-       створити файл із заданим ім’ям; якщо файл існує – перезаписати його вміст

-       написати код стійкий до різних варіантів вхідних параметрів

-       name – ім’я, може не включати шлях

-       записати у файл вміст стрічки content, прочитати content із стандартного вводу

-       повернути статус операції: Success – все пройшло успішно, Failure – файл не вдалося створити, або збій операції відкриття, запису даних, чи закриття файла.

**Задача №2 – Копіювання вмісту файла у інший файл**

***Реалізувати функцію створення файла і запису в нього даних:***

enum FileOpResult { Success, Failure, … };

FileOpResult copy\_file(char \*file\_from, char \*file\_to);

*Умови задачі:*

-       копіювати вміст файла з ім’ям file\_from у файл з ім’ям file\_to; написати код стійкий до різних варіантів вхідних параметрів, обробити всі можливі варіанти відсутності одного з файлів

-       file\_from, file\_to – можуть бути повним або відносним шляхом

-       повернути статус операції: Success – все пройшло успішно, Failure – файл не вдалося створити, або збій операції відкриття, читання чи запису даних, закриття файла.

**Self Practice Task:**

Вам дано N слів та число K.

Ваше завдання перечислити букви в словах, які зустрічаються в тексті більше-рівне ніж K разів (саме слово, не буква!).

Великі та маленькі букви вважаються однаковими, виводити необхідно малі, посортовані від останьої до першої у алфавіті. Букву потрібно виводити лише один раз.

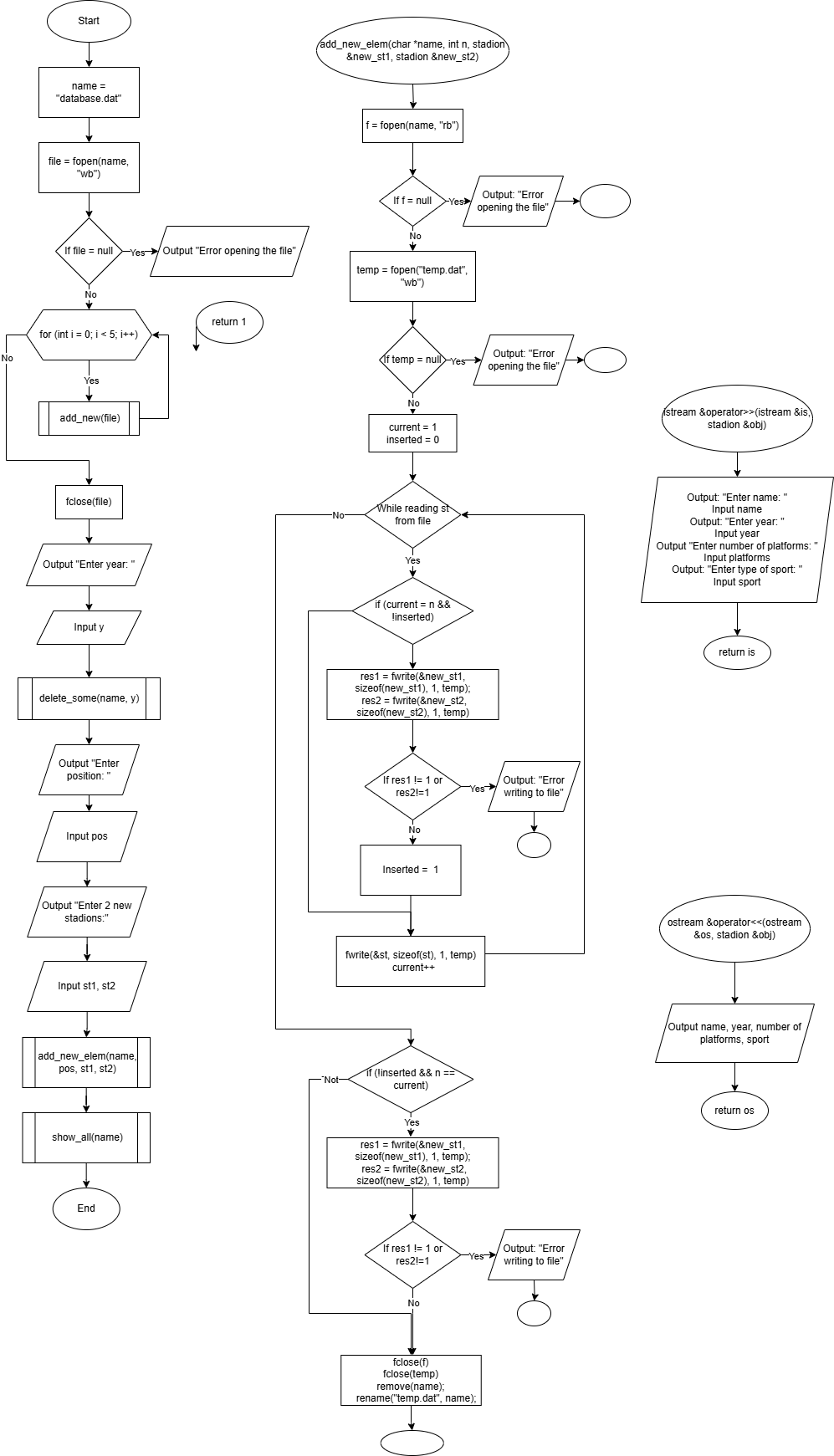
У випадку якщо таких букв немає - вивести "Empty!".

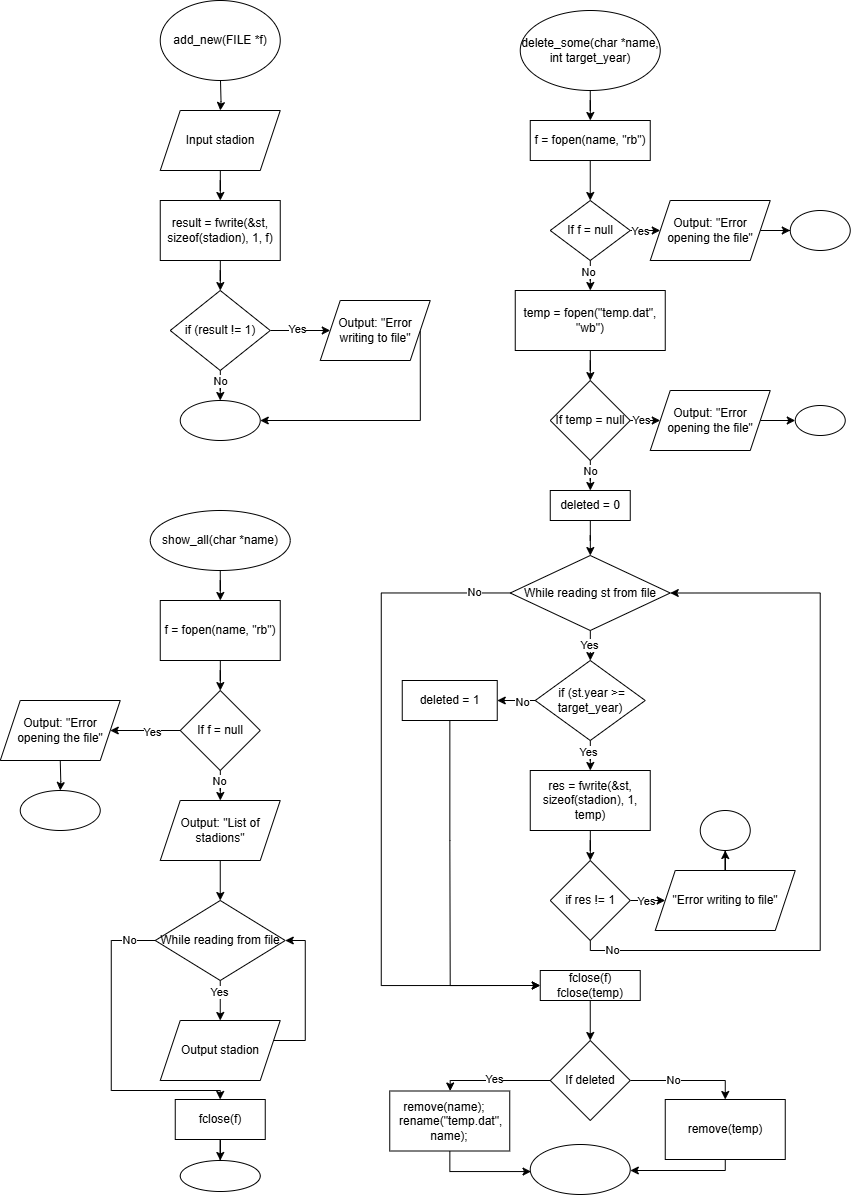
*2) Дизайн та планована оцінка часу виконання завдань*

**VNS Lab 6 Task 1 (23)**

Плановий час виконання – 30 хвилин.

**VNS Lab 8 Task 1 (23)**

****

****

Плановий час виконання – 2 години.

**VNS Lab 9 Task 1 (23)**

Плановий час виконання – 40 хвилин.

**Algotester Lab 4 Variant 1**

Плановий час виконання – 30 хвилин.

**Algotester Lab 4 Variant 1**

Плановий час виконання – 50 хвилин.

**Algotester Lab 6 Variant 2**

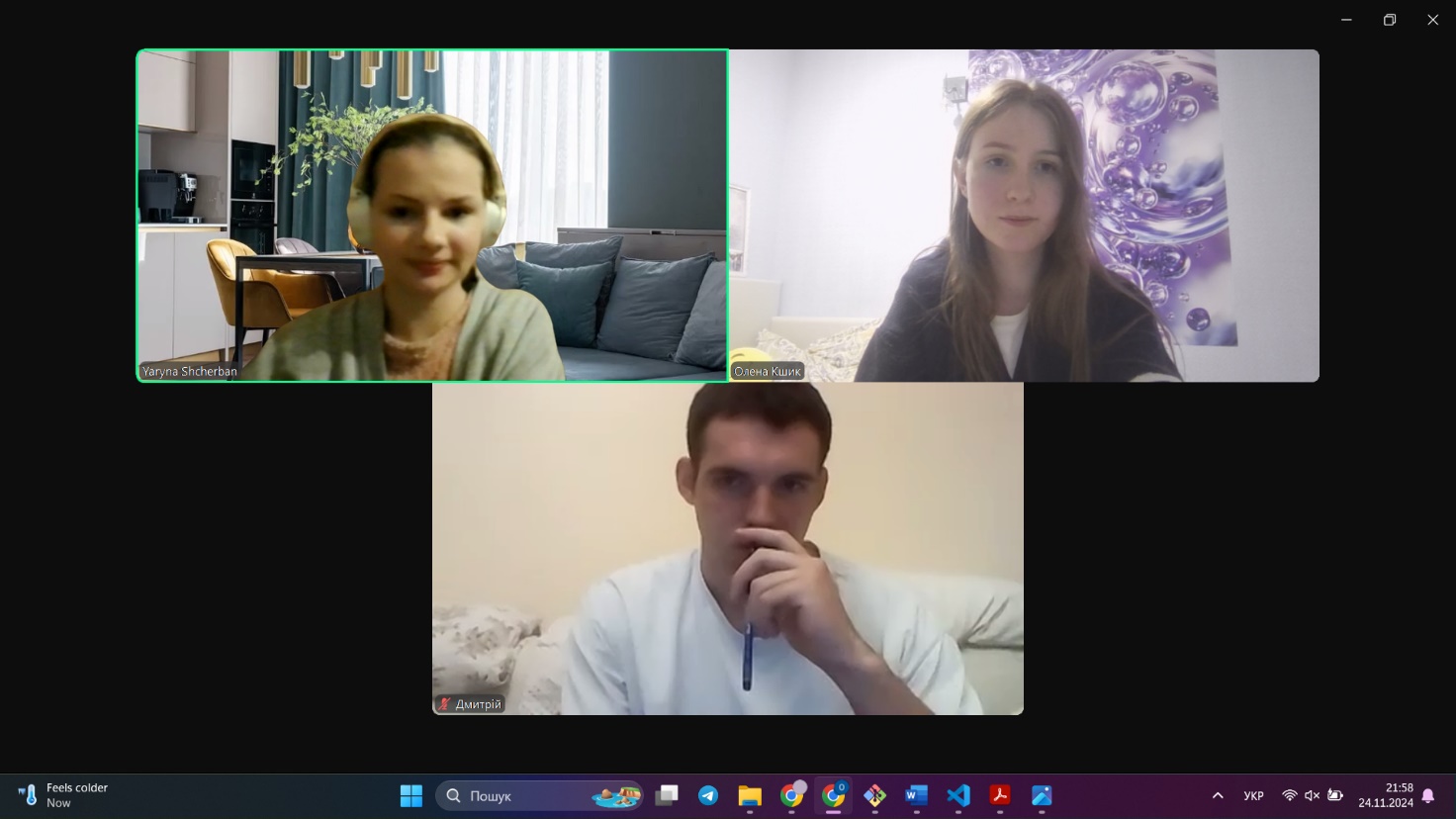
Плановий час виконання – 1 година.

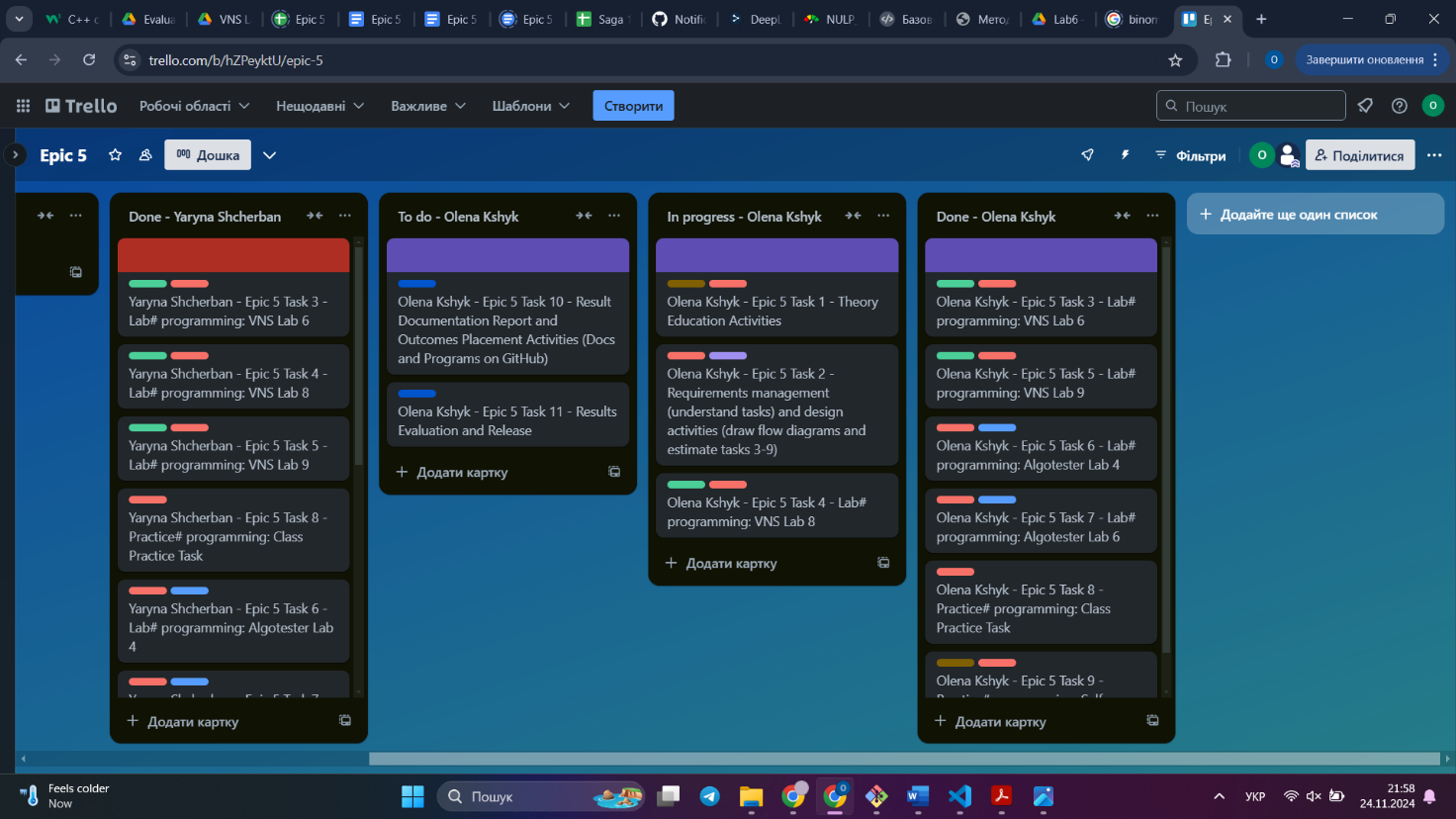
**Class Practice Work**

Плановий час виконання – 40 хвилин.

**Self Practice Task**

Плановий час виконання – 20 хвилин.

*3) Конфігурація середовища для виконання завдань*

**

*4) Код програми з посиланням на зовнішні ресурси*

VNS Lab 6 Task 1: **vns\_lab\_6\_task\_1\_variant\_23\_olena\_kshyk.cpp**

VNS Lab 8 Task 1: **vns\_lab\_8\_task\_1\_ variant\_23\_olena\_kshyk.cpp**

VNS Lab 9 Task 1: **vns\_lab\_9\_task\_1\_ variant\_23\_olena\_kshyk.cpp**

Algotester Lab 4 Variant 1: **algotester\_lab\_4\_task\_1\_olena\_kshyk.cpp**

Algotester Lab 4 Variant 1: **algotester\_lab\_4\_task\_1 \_variant\_2\_olena\_kshyk.cpp**

Class Practice Work: **practice\_work\_team\_tasks\_olena\_kshyk.cpp**

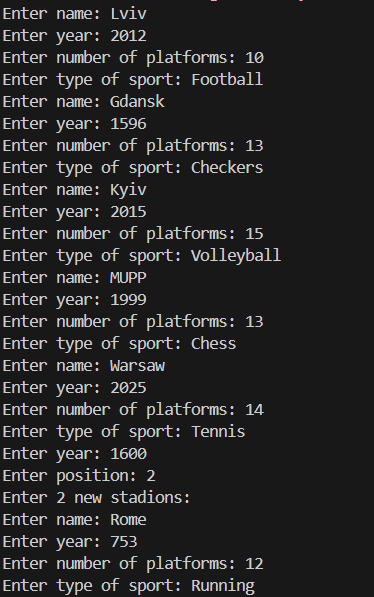
Self Practice Task: **practice\_work\_self\_algotester\_tasks\_olena\_kshyk.cpp**

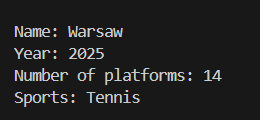
*5) Результати виконаних завдань, тестування та фактично затрачений час*

**VNS Lab 6 Task 1**

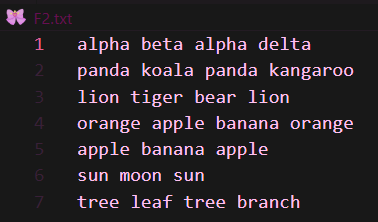
****

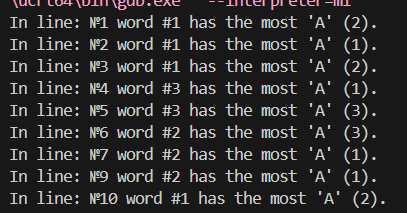
Фактичний час виконання – 1 година.

**VNS Lab 8 Task 1**

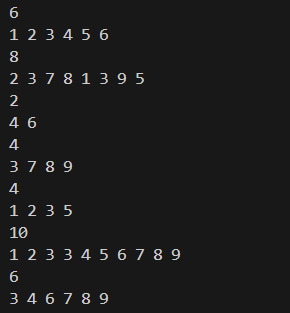
****

Фактичний час виконання – 4 години.

**VNS Lab 9 Task 1**

****

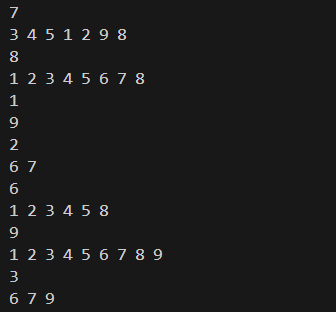
Фактичний час виконання – 40 хв.

**Algotester Lab 4 Variant 1**

Фактичний час виконання – 30 хвилин.

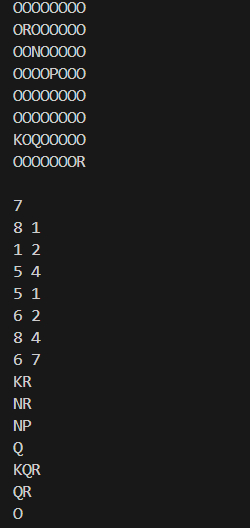
****

**Algotester Lab 4 Variant 1**

****Фактичний час виконання – 1.5 години.

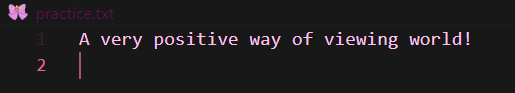
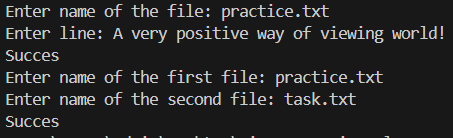
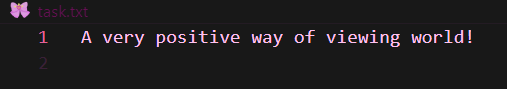
****

**Algotester Lab 6 Variant 2**

****

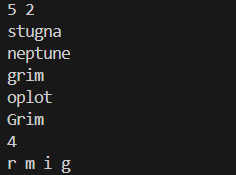
Фактичний час виконання – 2 години.

****

******Class Practice Task**

Фактичний час виконання – 40 хвилин.

**Self Practice Task**

Фактичний час виконання – 50 хвилин.

**Висновки:** У ході лабораторної роботи ми навчилися працювати з текстовими файлами, використовуючи стандартну бібліотеку мови C. Було розглянуто основні методи роботи з файлами: відкриття, читання, запис, перемотування, а також закриття файлів.