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 № 5**

На тему: «Програмування: алгоритм, програма, код. Системи числення. Двійкова система числення. Розробка та середовище розробки програми.»

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

до:

Практичних Робіт до блоку № 5

**Виконав:**

Студент групи ШІ-11

Левченко Денис

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:** Ознайомитися з основними поняттями роботи з файлами, включаючи бінарні та текстові файли, символи й рядкові змінні, а також засвоїти методи роботи з файлами за допомогою стандартної бібліотеки, створення та використання власних бібліотек.

**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:**

Вступ до Роботи з Файлами

Символи і Рядкові Змінні

Текстові Файли

Бінарні Файли

Стандартна бібліотека та робота з файлами

Створення й використання бібліотек

1. **Джерела**:

* Chat gpt
* Список відтворення на YouTube ( <https://youtube.com/playlist?list=PLiPRE8VmJzOpn6PzYf0higmCEyGzo2A5g&si=sXvmPdnGkwvJLXUi> )

**Виконання роботи:**

**VNS Lab 6 - Task 1-16:**

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.

Визначити які слова зустрічаються в рядку по одному разі.

**VNS Lab 8 - Task 1-15:**

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.

Структура "Власник автомобіля":

- прізвище, ім'я, по батькові;

- номер автомобіля;

- телефон;

- номер техпаспорта.

Знищити елемент із заданим номером, додати 2 елементи перед елементом із

заданим прізвищем.

**VNS Lab 9 - Task 1-16:**

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

інформацію

1) Скопіювати з файлу F1 у файл F2 всі рядки, що починаються на букву «А» і

закінчуються на букву «З», розташовані між рядками з номерами N1 й N2.

2) Визначити кількість слів у першому рядку файлу F2.

**Algotester Lab 4 v3:**

Вам дано масив, який складається з NN додатніх цілих чисел.

Ваше завдання - розділити його на три частини, по остачі від ділення на 3, по зростанню остачі (тобто спочатку йдуть числа, у яких остача 0, далі числа з остачею 1 і тоді нарешті числа з остачею 2).

Далі необхідно ті елементи, остача від ділення на 3 яких парна посортувати по зростанню, а ті, у яких остача 1 - по спаданню.

Після цього видаліть усі дублікати з масиву.

Виведіть результуючий масив.

# Вхідні дані

У першому рядку N - кількість чисел.

У другому рядку N чисел ai - елементи масиву.

# Вихідні дані

У першому рядку M - кількість чисел у масиву

У другому рядку M посоротованих за умовою чисел.

**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 4 v2:**

Вам дано масив a з N цілих чисел.  
Спочатку видаліть масиву aa усі елементи що повторюються, наприклад масив [1, 3, 3, 4] має перетворитися у [1, 3, 4].  
Після цього оберніть посортовану версію масиву aa на K, тобто при K=3K=3 масив [1, 2, 3, 4, 5, 6, 7] перетвориться на [4, 5, 6, 7, 1, 2, 3].  
Виведіть результат.

# Вхідні дані

У першому рядку цілі числа N та K

У другому рядку N цілих чисел - елементи масиву aa

# Вихідні дані

У першому рядку ціле число N - розмір множини aa  
У наступному рядку N цілих чисел - множина a

**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:**

У вас є шахова дошка розміром 8×88×8 та дуже багато фігур.

Кожна клітинка може мати таке значення:

* Пуста клітинка OO
* Пішак PP
* Тура RR
* Кінь NN
* Слон BB
* Король KK
* Королева QQ

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

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

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

У випадку, якщо клітинку не атакують - виведіть OO.

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.

**Class Practice Work:**

Реалізувати функцію створення файла і запису в нього даних:

enum FileOpResult { Success, Failure, … };

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

Реалізувати функцію створення файла і запису в нього даних:

enum FileOpResult { Success, Failure, … };

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

**Self Practice Task (6.1):**

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,

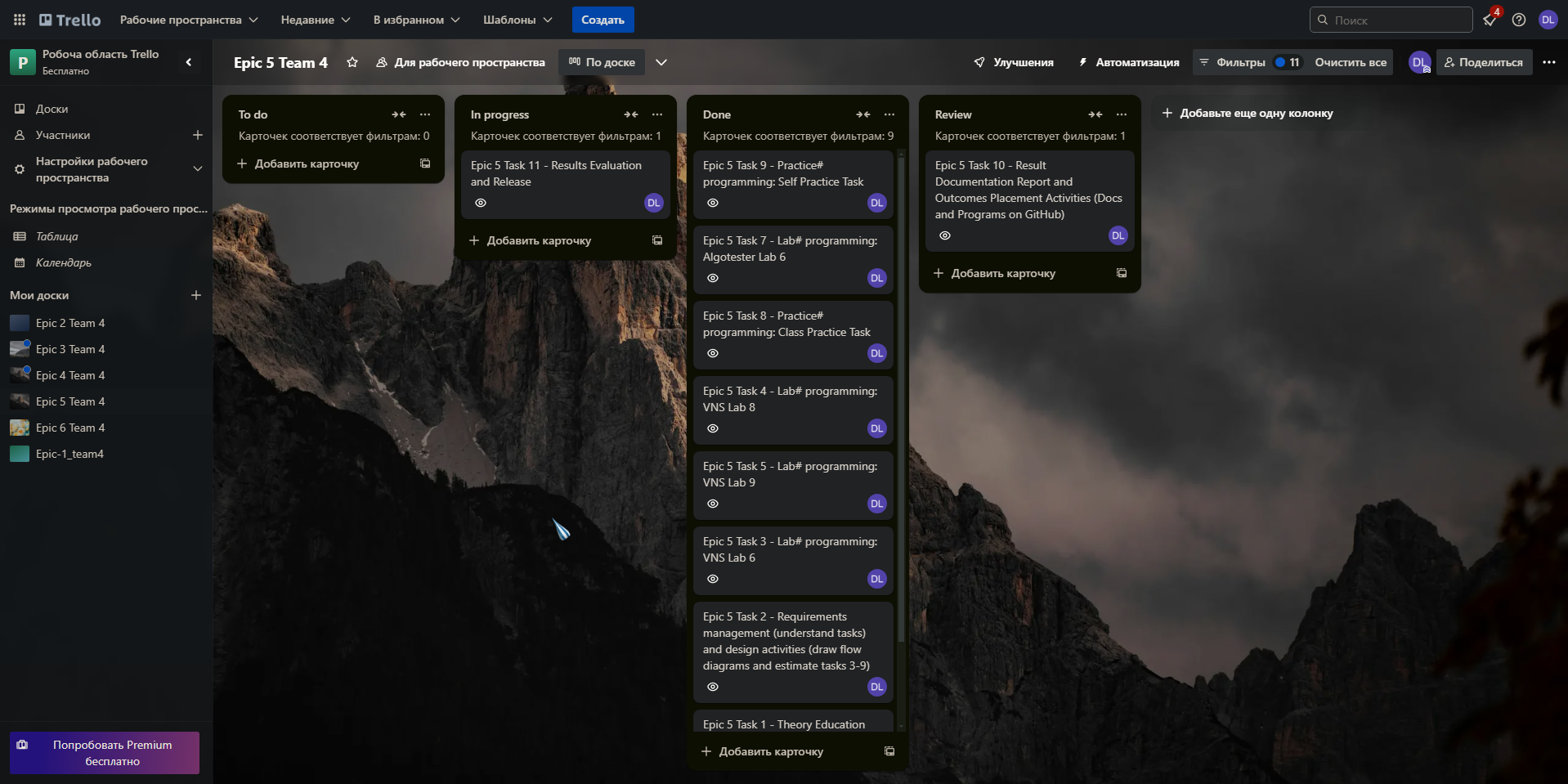
посортовані від останьої до першої у алфавіті. Букву потрібно виводити лише

один раз.

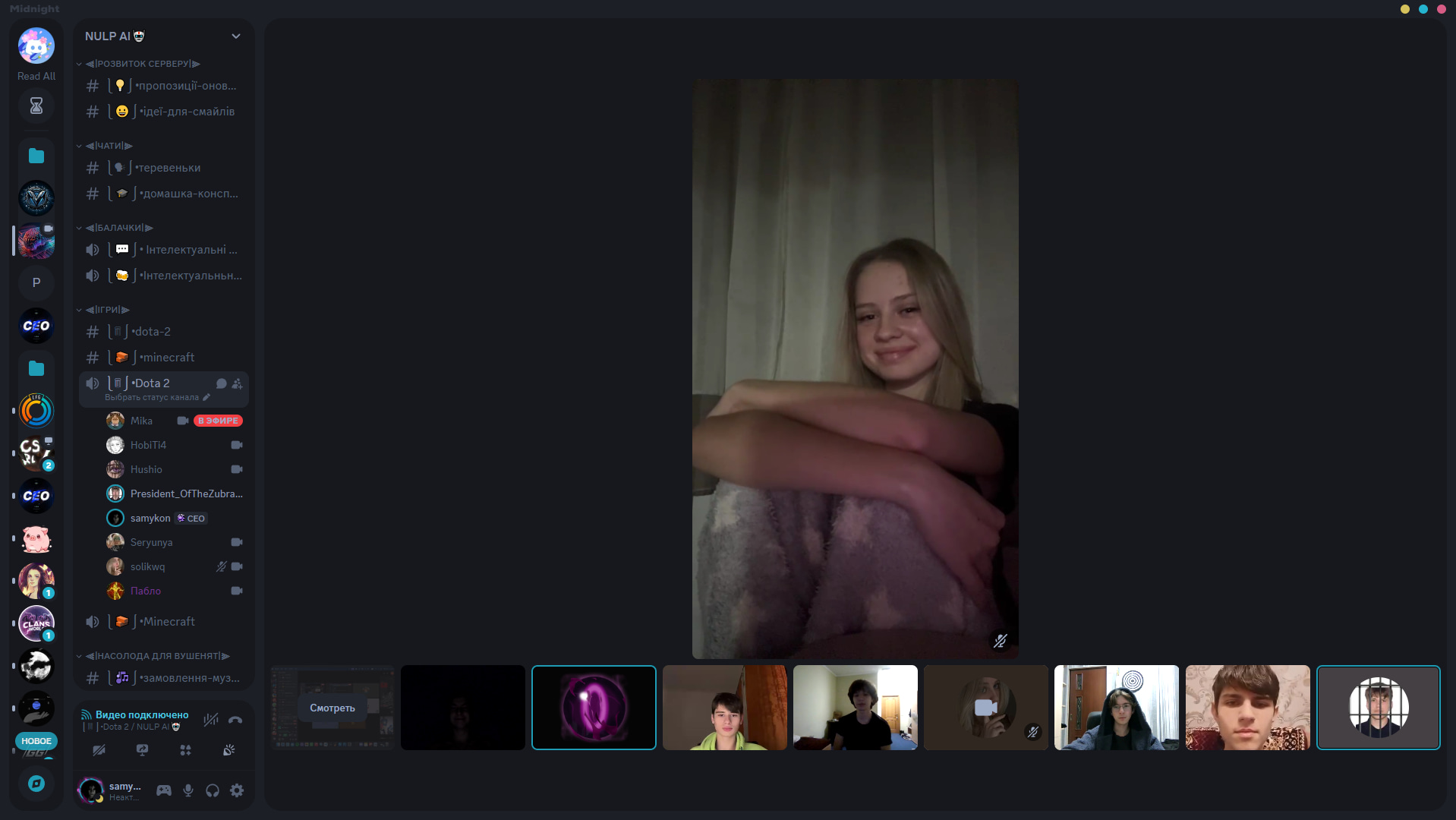
У випадку якщо таких букв немає - вивести "Empty!".

1. Requirements management and design activities

Team Trello dashboard for task control



Team meeting



**Time planning for tasks & tasks design**

VNS Lab 6

*~20хв*

VNS Lab 8

*~3год*

VNS Lab 9

*~30хв*

Algotester Lab 4

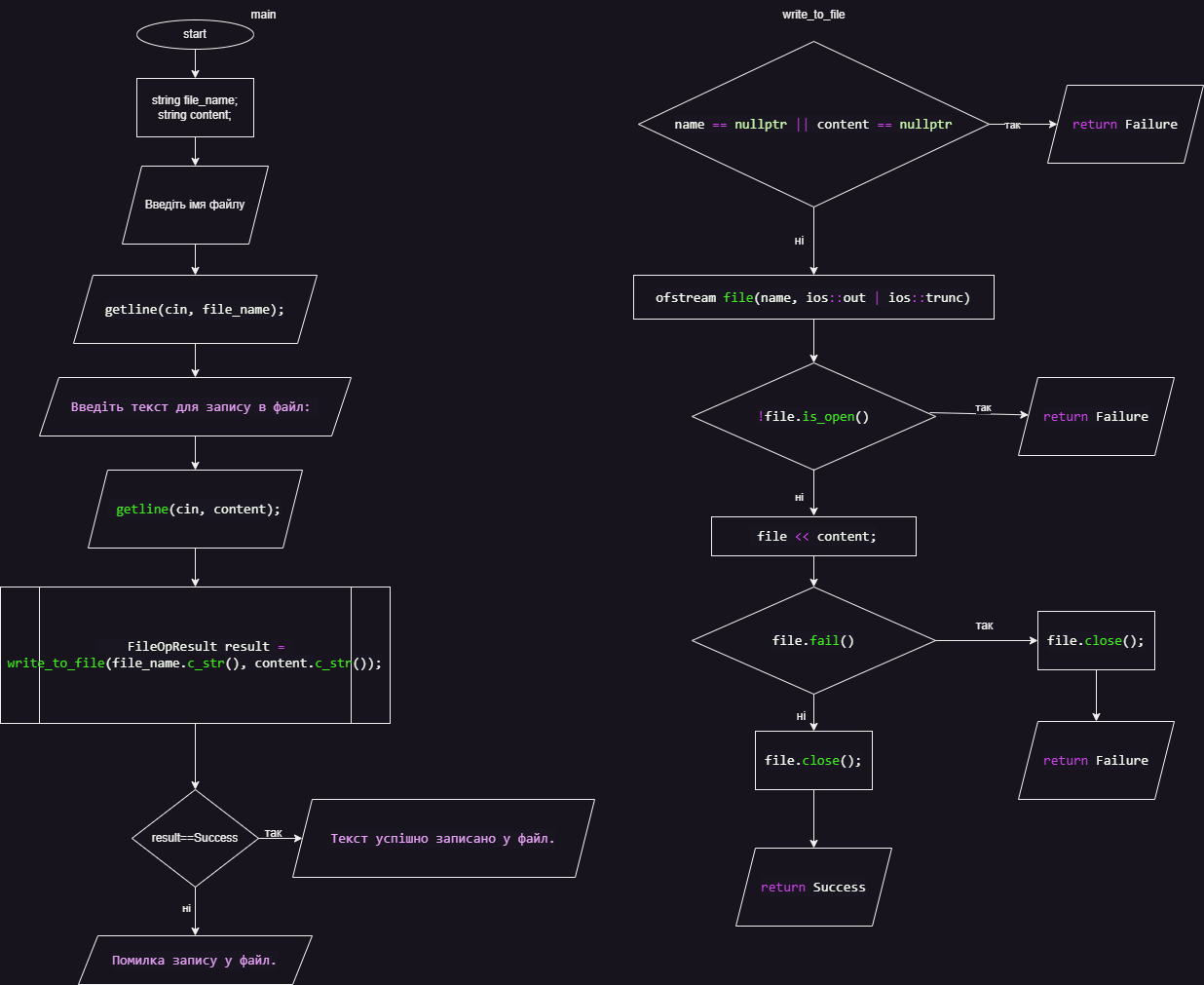
*~10хв*

Algotester Lab 4

*~10хв*

Class Practice Work

*~50хв*

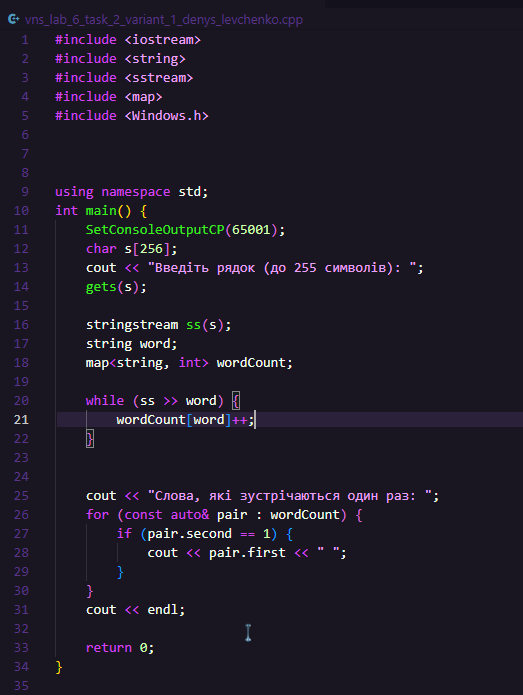


Self Practice Work

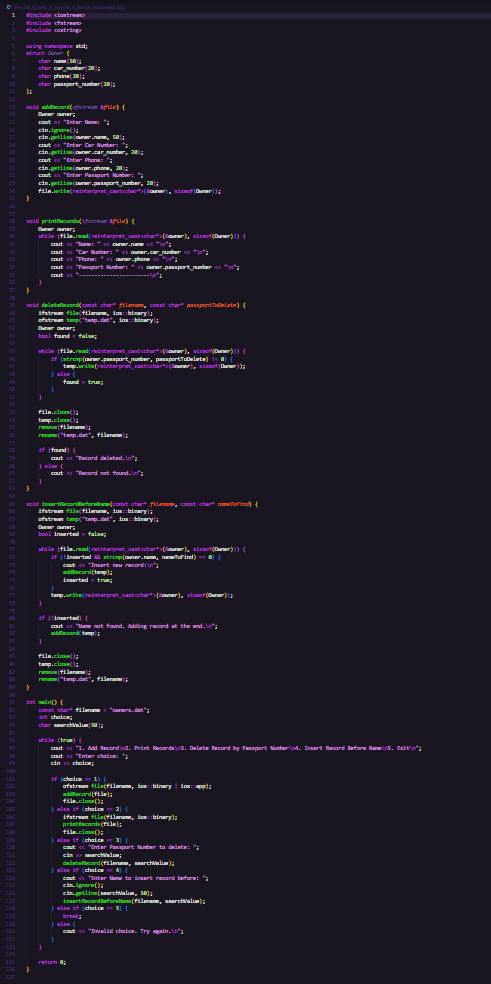
*~30хв*

Код програми з посиланням на зовнішні ресурси

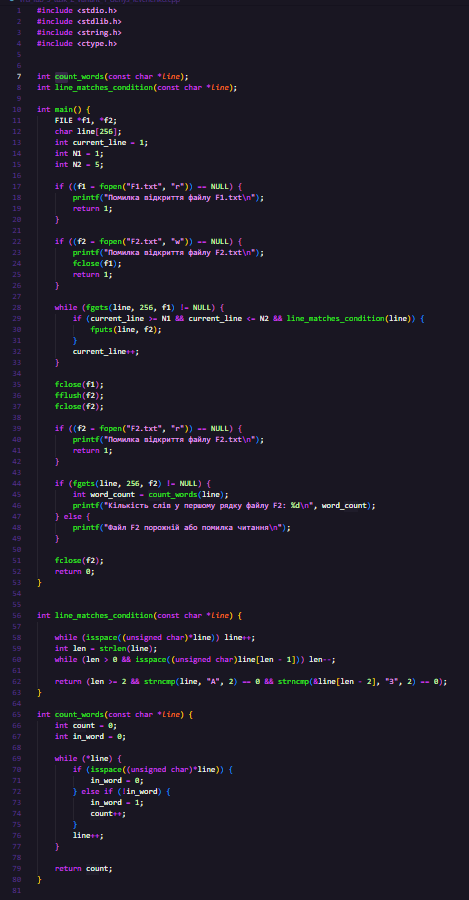
VNS Lab 6



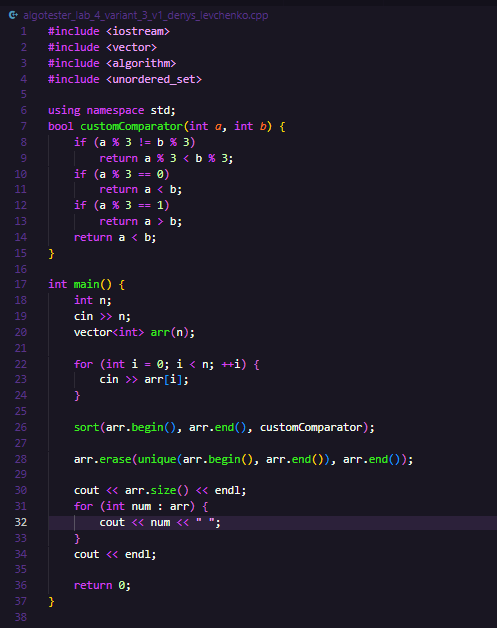
VNS Lab 8:



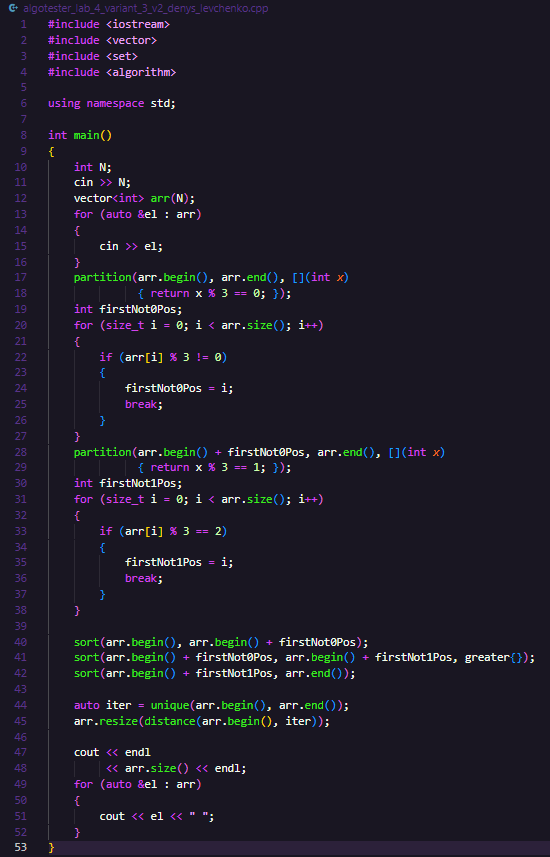
VNS Lab 9



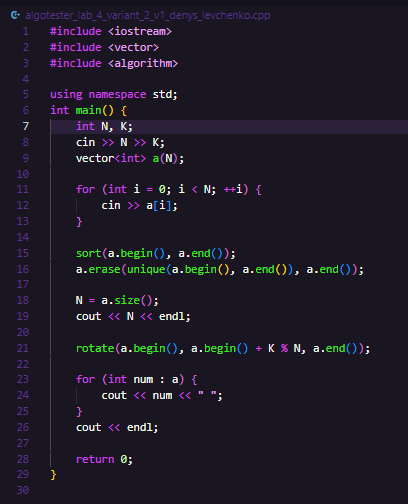
Algotester Lab 4 variant 3:



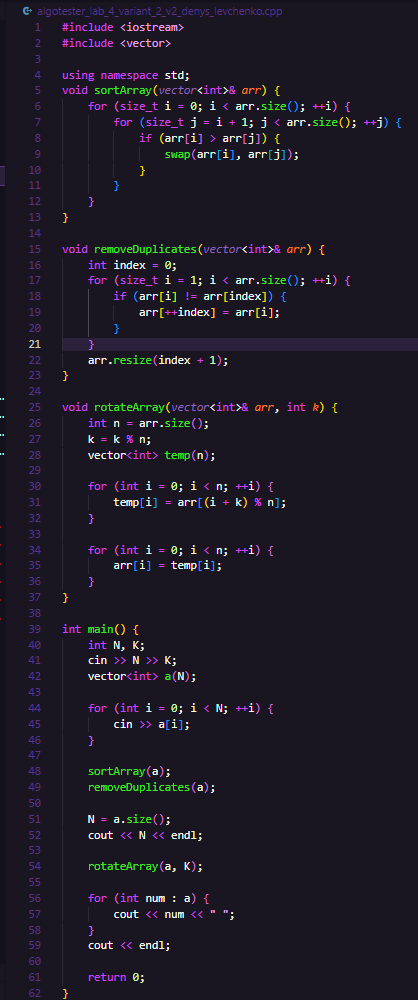
Algotester Lab 4 variant 3 v2:



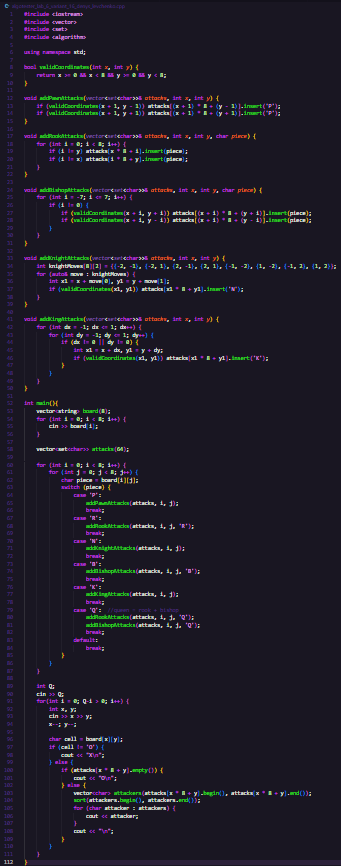
Algotester Lab 4 variant 2:



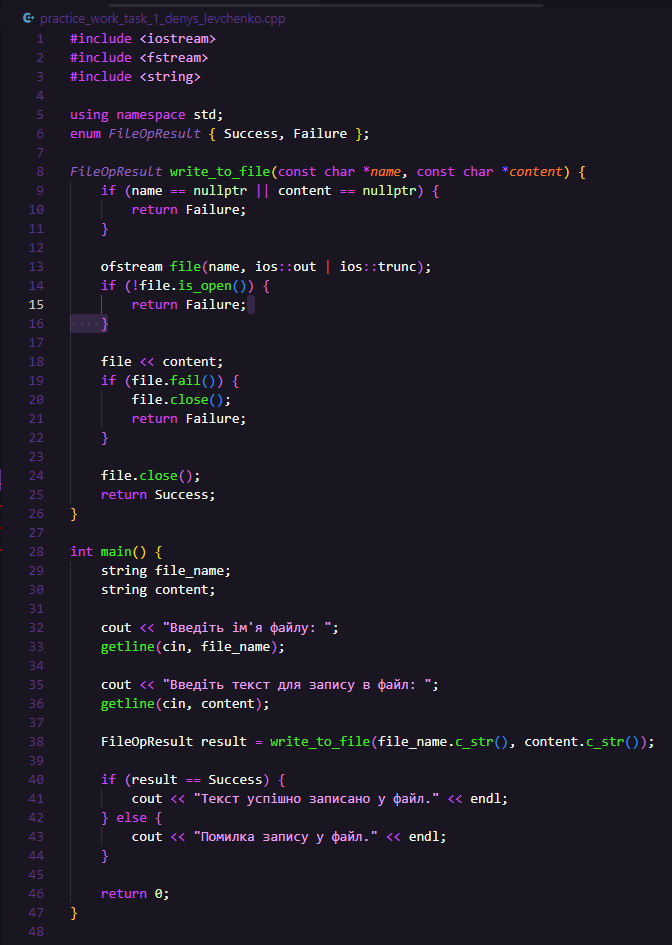
Algotester Lab 4 variant 2 v2:



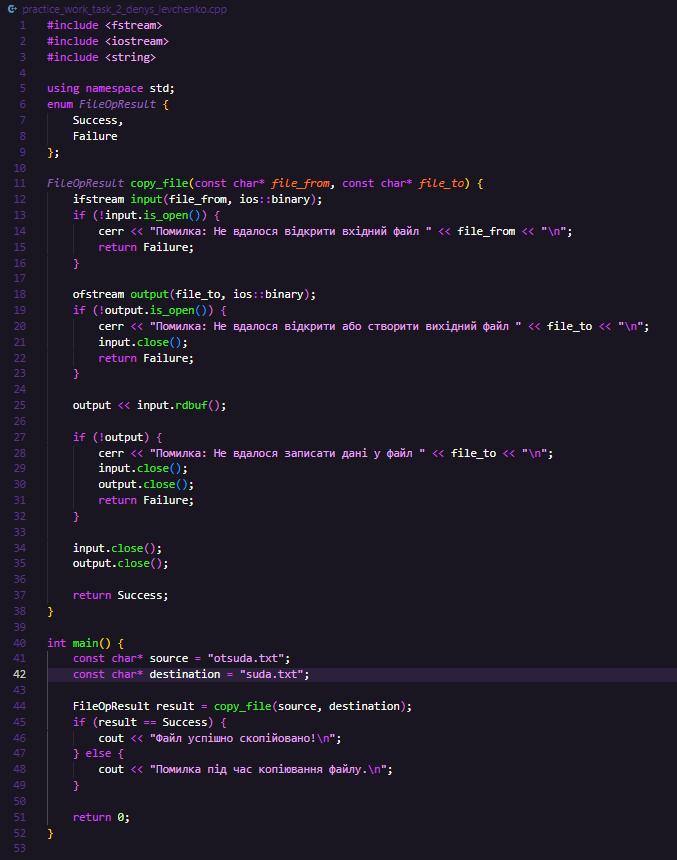
Algotester Lab 6 :



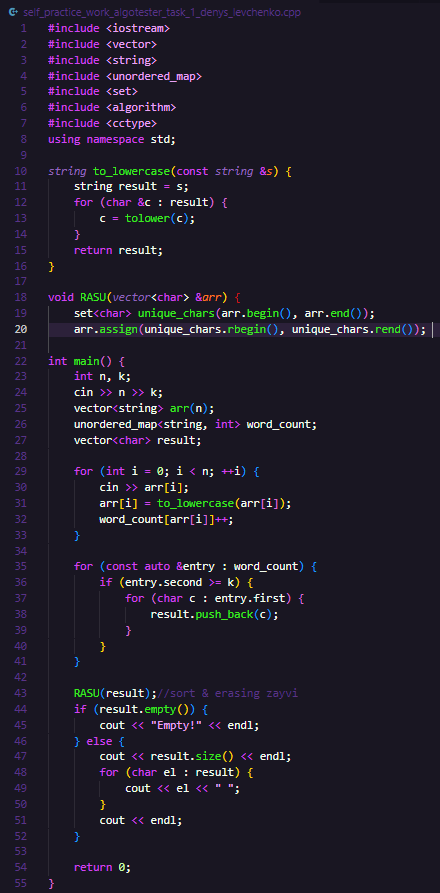
Class Practice Task 1:



Class Practice Task 2:



Self Practice Task:



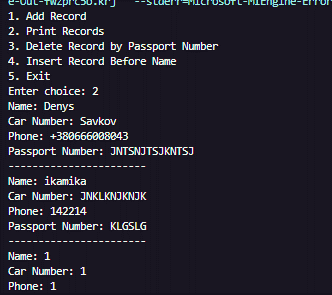
**Результати виконаних завдань, тестування та фактично затрачений час**

VNS Lab 6



Затратність ~35хв

VNS Lab 8



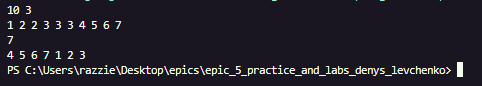
Затратність ~15год

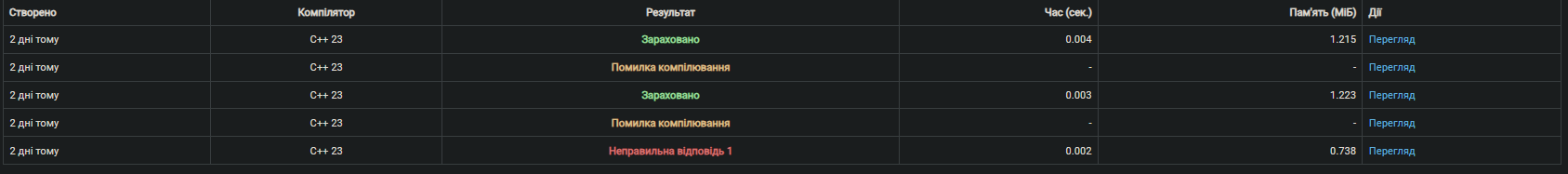
VNS lab 9



Затратність ~1год

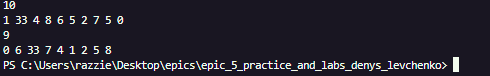
Algotester Lab 4 variant 2

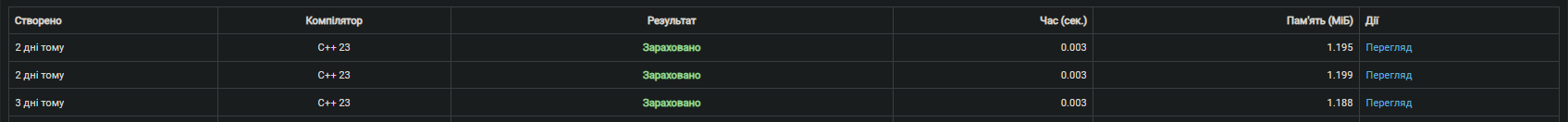




Затратність ~3год

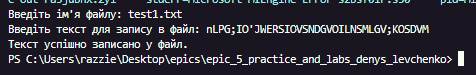
Algotester Lab 4 variant 3





Затратність ~3год

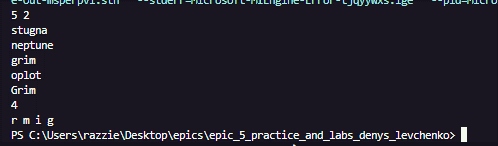
Class Practice Task





Затратність ~40хв

Self Practice Task





Затратність ~1год

**Висновки:**

Я навчився працювати з файлами різних типів, включаючи бінарні та текстові, використовуючи символи й рядкові змінні. Завдяки стандартній бібліотеці та створенню власних бібліотек, я отримав практичні навички ефективного збереження, обробки та взаємодії з даними.