Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра систем штучного інтелекту



**Звіт**

про виконання

**Лабораторних та практичних робіт № 4,5**

***з дисципліни:*** «Мови та парадигми програмування»

***з розділу***: «Прості структури даних. Одновимірні масиви. Двовимірні Масиви. Алгоритми обробки»

***Виконав(ла):***

студент групи ШІ-12

Ширяєв Іван

# **Тема роботи:**

Опрацювання масивів, їх різновидностей. Знайомство з принципами і алгоритмами обробки цих структур даних.

# **Мета роботи:**

Навчитись працювати з одновимірними і двовимірними масивами. Ознайомитись з імплементацією простих структур даних і алгоритмів їх обробки для розв’язання задач.

# **Теоретичні відомості:**

1. Теоретичні відомості з переліком важливих тем:

* Тема №1: Масиви.
* Тема №2: Двовимірні масиви.
* Тема №3: Обробка масивів

1. Індивідуальний план опрацювання теорії:

* Тема №1: Масиви.
  + Джерела Інформації
    - <https://www.w3schools.com/cpp/cpp_arrays.asp>
    - <https://www.geeksforgeeks.org/cpp-arrays/>
    - <https://cplusplus.com/doc/tutorial/arrays/>
  + Що опрацьовано:
    - Безпосередньо опрацьовано масиви, їх різновидності, базові імплементацію та роботу з ними.
  + Статус: Ознайомлений
  + Початок опрацювання теми: 1.11.2023
  + Звершення опрацювання теми: 4.11.2023
* Тема №2: Двовимірні масиви.
  + Джерела Інформації:
    - <https://www.geeksforgeeks.org/cpp-multidimensional-array/>
    - <https://www.digitalocean.com/community/tutorials/two-dimensional-array-in-c-plus-plus>
  + Що опрацьовано:
    - Опрацьовано особливості і імплементацію двовимірних масивів.
  + Статус: Ознайомлений
  + Початок опрацювання теми: 15.11.2023
  + Звершення опрацювання теми: 16.11.2023
* Тема №3: Обробка масивів
  + Джерела Інформації:
    - все в попередніх джерелах
  + Що опрацьовано:
    - опрацьована обробка масивів, деякі алгоритми для цього.
  + Статус: Ознайомлений
  + Початок опрацювання теми: 20.11.2023
  + Звершення опрацювання теми: 25.11.2023

# **Виконання роботи:**

## **1. Опрацювання завдання та вимог до програм та середовища:**

Завдання VNS lab 4 Task 1-1

* Варіант 1
* Деталі завдання

1) Сформувати одновимірний масив цілих чисел, використовуючи генератор

випадкових чисел.

2) Роздрукувати отриманий масив.

3) Знищити елемент із номером К.

4) Додати після кожного парного елемента масиву елемент зі значенням 0.

5) Роздрукувати отриманий масив.

* Важливі деталі для врахування в імплементації програми

Використовуємо <random> та <time.h> для імплементації рандому.

Завдання VNS lab 5 Task 1-1

* Варіант 1
* Деталі завдання

У двовимірному масиві записані слова, що представляють собою

послідовність цифр, що завершуються 0. Необхідно роздрукувати слова через

кому, взявши надрукований рядок у дужки.

* Важливі деталі для врахування в імплементації програми

Добування слова оформити у вигляді функції.

Завдання Algotester 2

* Варіант 1
* Деталі завдання

У вас є дорога, яка виглядає як N чисел.

Після того як ви по ній пройдете - вашу втому можна визначити як різницю максимального та мінімального елементу.

Ви хочете мінімізувати втому, але все що ви можете зробити - викинути одне число з дороги, тобто забрати його з масиву.

В результаті цієї дії, яку мінімальну втому ви можете отримати в кінці дороги?

* Важливі деталі для врахування в імплементації програми

Потрібно використати алгоритм сортування.

Завдання Algotester 3

* Варіант 2
* Деталі завдання

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.

* Важливі деталі для врахування в імплементації програми

Немає

Завдання Class practice work

* Варіанту немає
* Деталі завдання

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

* Важливі деталі для врахування в імплементації програми

Використовуємо функцію stoi() для конвертації рядка в число.

Завдання Self practice work

* Варіанту немає
* Деталі завдання

Мале Бісеня любить гострити зуби. А Зла Тітонька любить до нього підходити і питатися: «Що, зуби гостриш?». Бісеняті таке не дуже подобається, тому воно придумало робити таке.

У Малого Бісеняти є n зубів. Кожен зуб має коефіцієнт загостреності ai. Також існує межа загостреності k. Якщо коефіцієнт загостреності певного зуба є більшим чи рівним межі загостреності, то такий зуб вважається загостреним.

Мале Бісеня хоче наступного разу, коли Зла Тітонька його щось запитає, показати їй якнайбільше загострених зубів, що розташовані поспіль.

* Важливі деталі для врахування в імплементації програми

Немає

## **2. Дизайн та планована оцінка часу виконання завдань:**

Програма VNS lab 4 Task 1-1

* Планований час на реалізацію: 1 година

Програма VNS lab 5 Task 1-1

* Планований час на реалізацію: 1 година

Програма Algotester 2

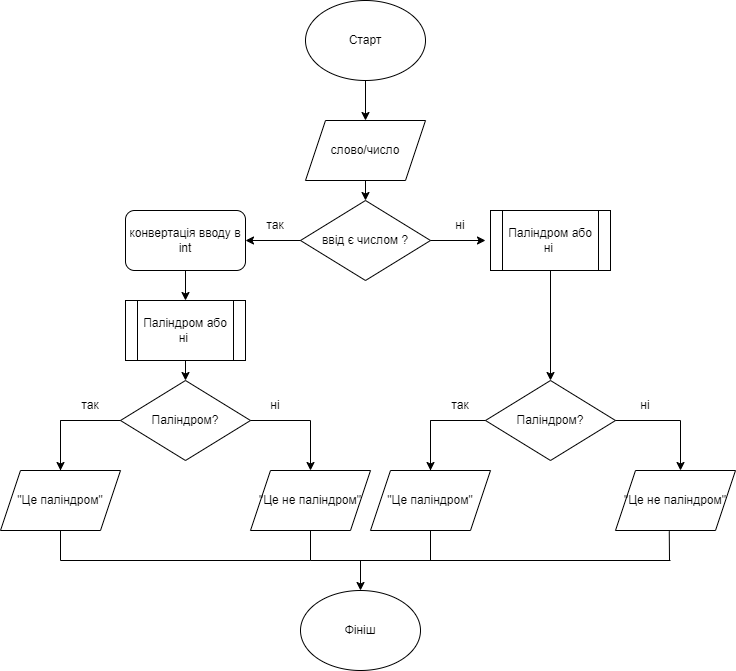
* Планований час на реалізацію: 1 година

Програма Algotester 3

* Планований час на реалізацію 2 години

Програма Class practice work

* Блок-схема



*Блок схема Class practice work*

* Планований час на реалізацію: 1 година

Програма Self practice work

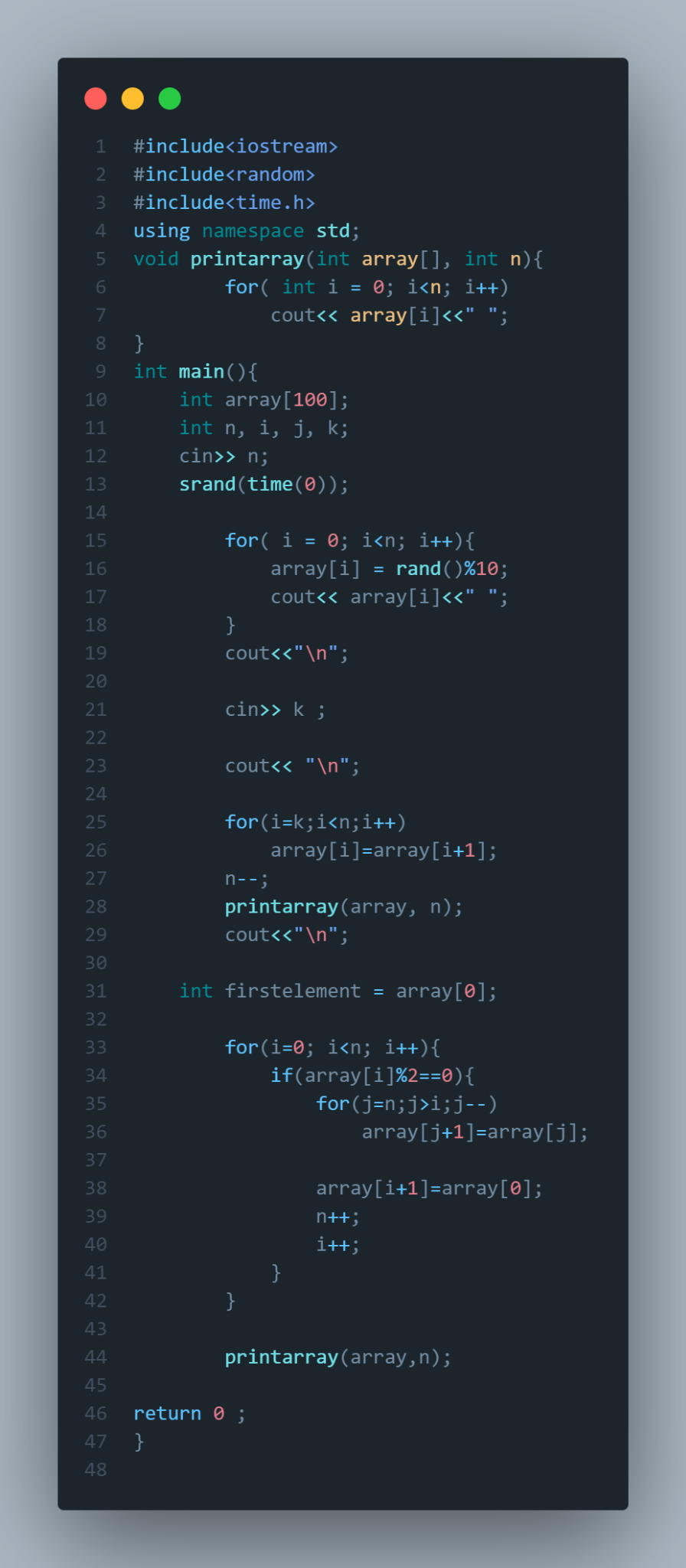
* Планований час на реалізацію: 1 година

## **3. Конфігурація середовища до виконання завдань:**

Додаткова конфігурація не потрібна.

## **4. Код програм з посиланням на зовнішні ресурси:**

Завдання VNS lab 4 Task 1-1 Деталі по програмі + Вставка з кодом з підписами до вставки. Посилання на файл програми у пул-запиті GitHub



*Код VNS lab 4*

Задаємо статичний масив з псевдозмінними межами, задаємо сід для функції рандому, заповнюємо масив випадковими значеннями від 0 до 9. Задаємо число К, зміщуємо всі елементи масиву після елементу під номером К ліворуч щоб “прибрати” цей елемент з масиву, зменшуємо його довжину на 1, після цього є цикл що проходиться по елементам масиву, перевіряє кожний на парність, елементи після кожного парного зміщуємо на 1 праворуч і в пусте місце вставляємо елемент під номером 0, виводимо масив.

Завдання VNS lab 5 Task 1-1



*Код VNS 5*

Задаємо двовимірний масив та функцію що буде проходитись по елементам кожного рядка і перевіряти чи є він нулем, кожний ненульовий елемент виводимо, замість кожного нуля виводимо кому, перед та після рядків ставимо дужки.

Завдання Algotester 2



*Код Algotester 2*

В завданні не зазначено чи обов’язково потрібно вводити значення масиву у збільшувальному порядку, тому я використав bubble sort для того щоб відсортувати значення для більш легкої роботи з ними. Нам потрібно “викинути” один елемент з масиву для того щоб знайти мінімальну втому, “викидувати” будемо тільки перший або останній елемент масиву бо тільки в цьому є сенс. Тобто розглядаємо два випадки: різниця першого і передостаннього елементу масиву і навпаки - другого і останнього, обчислюємо їх за допомогою функції min знаходимо менше з цих двох значень, виводимо його.

Завдання Algotester 3



*Код Algotester 3*

Вводимо масив з N значень, та масив з M значень, порівнюємо кожен елемент, якщо вони одинакові то збільшуємо лічильник на 1, це буде кількість одинакових елементів в обох масивах, після цього знаходимо різницю довжин масивів і лічильника - це кількість унікальних елементів

Завдання Class practice work



*Код Class practice work*

Вводимо число або слово яке бажаємо перевірити, дивимось чи це число або слово, якщо це число то конвертуємо ввід в інт, функція для чисел працює таким чином : ми розвертаємо число та дивимось чи дорівнює воно початковому, якщо так то це - паліндром, для слів ми користуємось рекурсією та попарно порівнюємо перший і останні елементи і так далі. В кінці виводимо чи є ввід паліндромом.

Завдання Self practice work

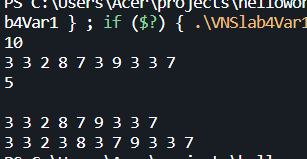


*Код Algotester Self Practice*

Вводимо кількість зубів та межу гостроти, вводимо вектор що зберігає значення гостороти зубів. Цикл порівнює значення в масиві і межу, якщо значення рівне або більше то збільшуємо лічильник на 1, і заносимо значення лічильнику до буферного вектору, якщо значення менше то обнуляємо його і також заносимо заносимо до вектору, після того як перевірили всі числа знаходимо максимальне значення в буферному векторі і виводимо його.

## **5. Результати виконання завдань, тестування та фактично затрачений час:**

Завдання VNS lab 4 Task 1-1 Деталі по виконанню і тестуванню програми



*Результат VNS lab 4*

Вводимо довжину масиву,, сам масив, вводимо номер елементу який бажаємо видалити,

масив без цього елементу, масив в якому після кожного парного елементу стоїть елемент під номером К.

Час затрачений на виконання завдання: 2 години

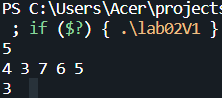
Завдання VNS lab 5 Task 1-1 Деталі по виконанню і тестуванню програми



*Результат VNS lab 5*

Час затрачений на виконання завдання: 40 хвилин

Завдання Algotester 2 Деталі по виконанню і тестуванню програми

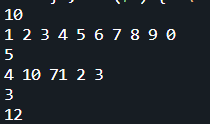


*Результат Algotester 2*



Час затрачений на виконання завдання: 2 години

Завдання Algotester 3 Деталі по виконанню і тестуванню програми



*Результат Algotester 3*

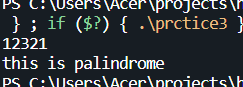
**

*Результат на алготестері*

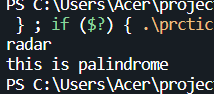
однакові елементи - 2, 3, 4.

Час затрачений на виконання завдання: 1.5 години

Завдання Class practice Деталі по виконанню і тестуванню програми



*Результат Class practice work*



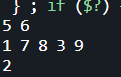
*Результат Class practice work*



*Результат Class practice work*

Час затрачений на виконання завдання: 2.5 години.

Завдання Self Practice Work Деталі по виконанню і тестуванню програми



*Результат Self practice work*

**

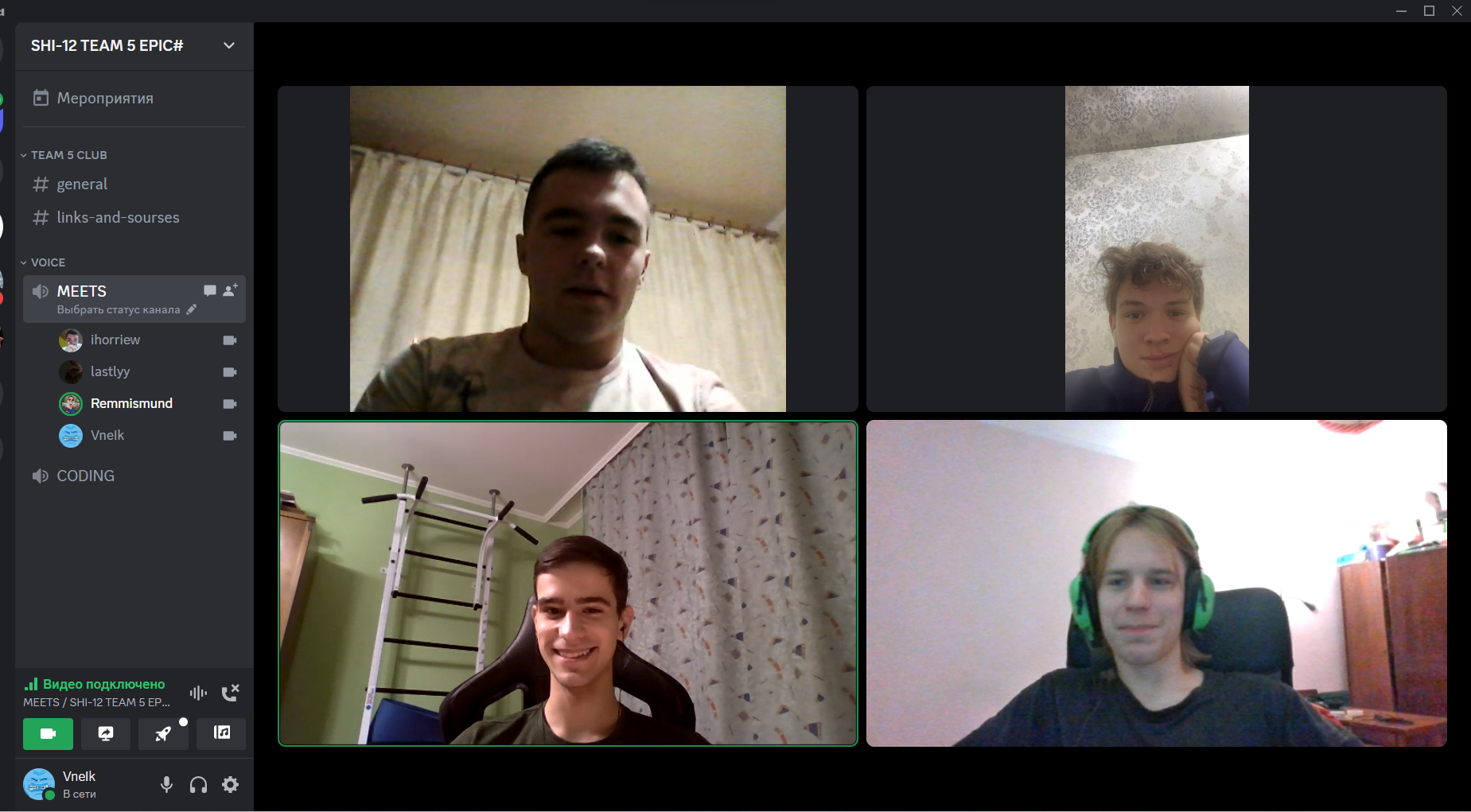
*Результат на Algotester*

тут найдовша послідовність - 7, 8.

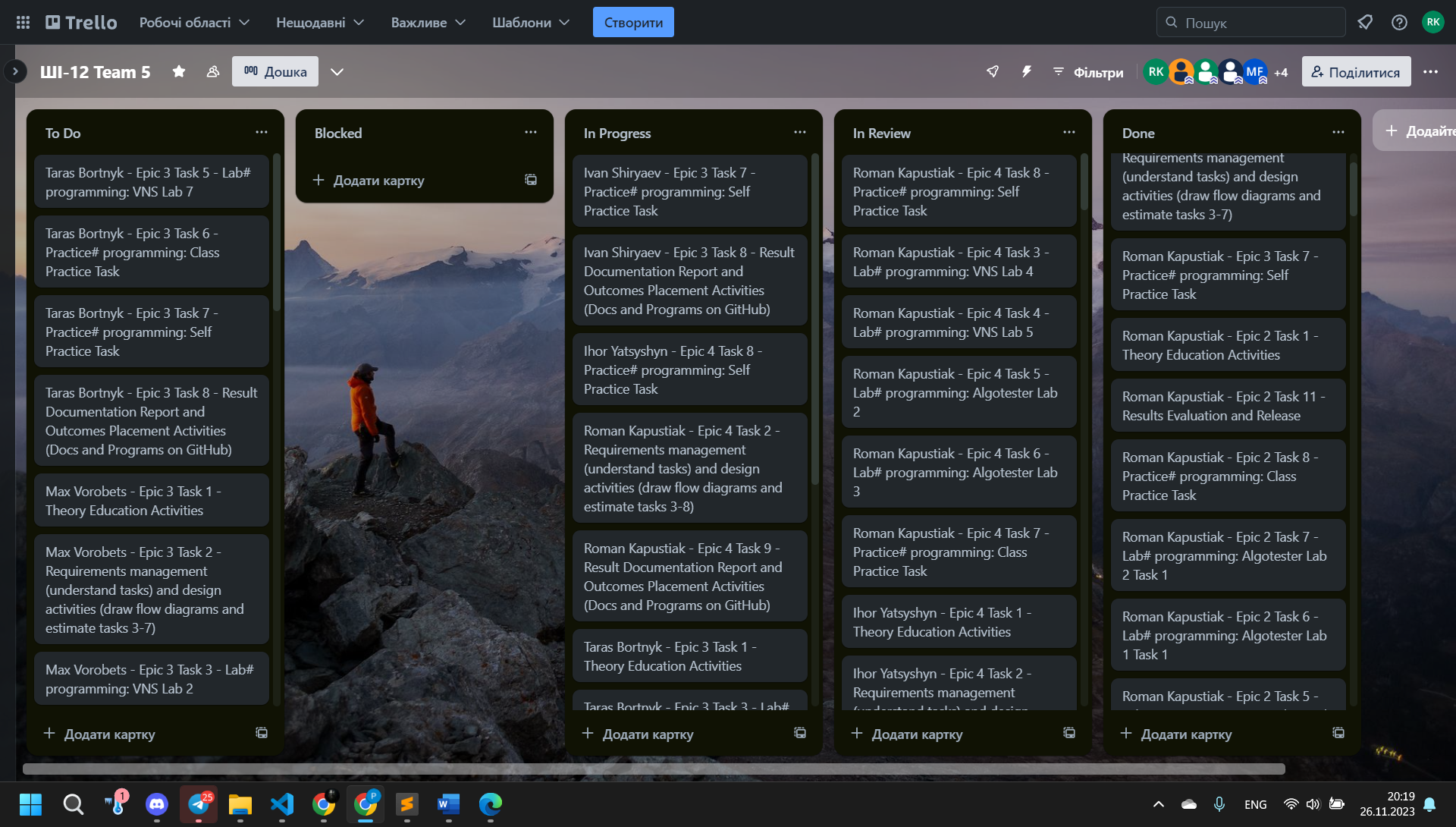
Час затрачений на виконання завдання: 1.5 години

## **6. Кооперація з командою:**

* Скрін з 1-ї зустрічі по обговоренню задач Епіку та Скрін прогресу по Трелло



*Перший міт*

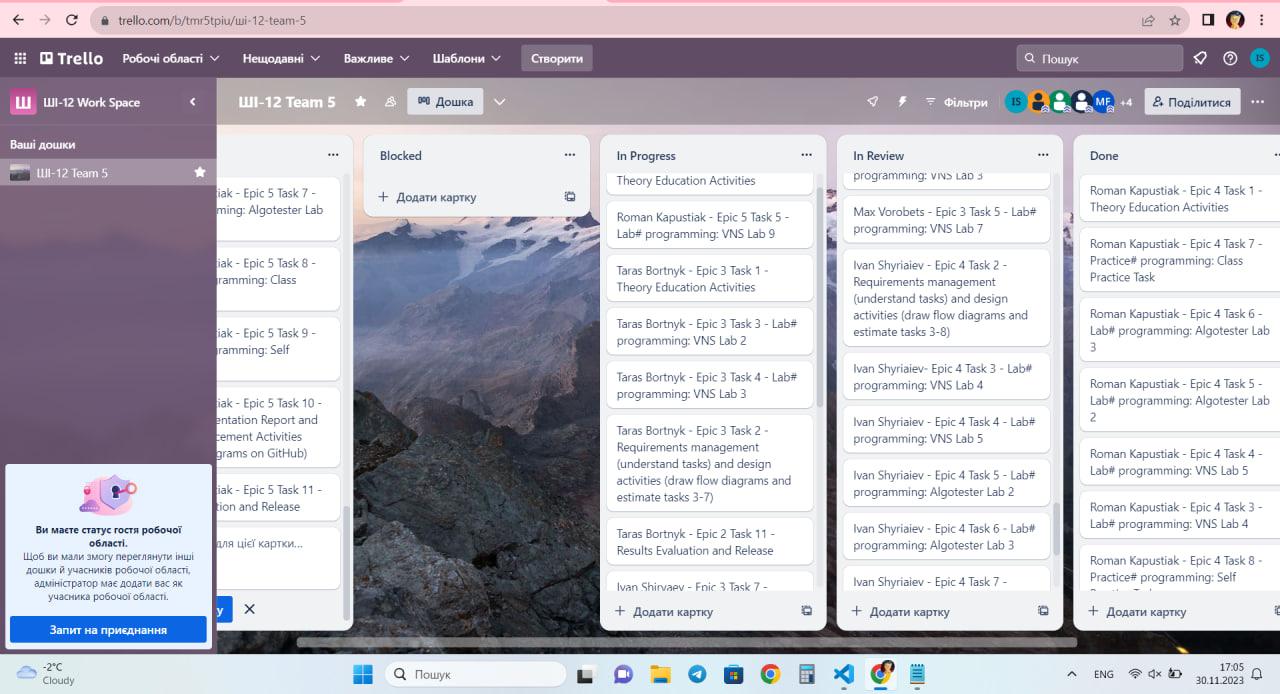


*Трелло прогрес*

* Скрін з 2-ї зустрічі по обговоренню задач Епіку та Скрін прогресу по Трелло



*Другий міт*



*Трелло прогрес*

* Скрін з 2-му коментарями від учасників команди на пул реквесті з Ревю Роботи

# **Висновки:**

Під час цієї роботи я ознайомився з такими поняттями як масив, двовимірний масив. Я навчився використовувати їх для розв’язання задач, також я дізнався як можна імплементувати рандом в C++ та конвертувати стрічки в число.