Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра систем штучного інтелекту



**Звіт**

про виконання

**Лабораторних та практичних робіт № 4**

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

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

***Виконала:***

студентка групи ШІ-11

Савчук Аня Григорівна

# **Тема роботи:**

# Опрацювання теоретичного матеріалу за темами. Застосування вивченого матеріалу на практиці. Прості структури даних. Одновимірні масиви. Двовимірні Масиви. Алгоритми обробки.

# **Мета роботи:**

Опрацювати теорію та навчитись практично застосовувати отримані знання. Виконати практичне завдання та лабораторні роботи згідно свого варіанту.

# **Теоретичні відомості:**

1. Теоретичні відомості з переліком важливих тем:

* Тема №1: Одновимірні масиви.
* Тема №2: Двовимірні Масиви.

1. Індивідуальний план опрацювання теорії:

* Тема №1: Одновимірні масиви.
  + Джерела Інформації
    - <https://acode.com.ua/urok-77-masyvy/>
    - <https://www.youtube.com/watch?v=ULdbOaMBPYc>
    - <http://cpp.dp.ua/obrobka-odnovymirnyh-masyviv/>
  + Що опрацьовано:
    - Поняття масиву. Елементи масиву. Оголошення масивів фіксованого розміру. Синтаксис створення масиву. Робота з одновимірними масивами.
  + Статус: Ознайомлений
  + Початок опрацювання теми: 01.11.23
  + Звершення опрацювання теми: 08.11.23
* Тема №2: Двовимірні Масиви.
  + Джерела Інформації:
    - <https://acode.com.ua/urok-81-bagatovymirni-masyvy/>
    - <https://www.youtube.com/watch?v=V2g3B9Zbh4Q>
  + Що опрацьовано:
    - Багатовимірін масиви. Ініціалізація двовимірного масиву. Доступ до елементів в двовимірному масиві. Робота з двовимірними масивами
  + Статус: Ознайомлений
  + Початок опрацювання теми: 08.11.23
  + Звершення опрацювання теми: 15.11.23

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

Завдання №1 **VNS Lab 4**

* Варіант завдання: **5**
* Деталі завдання:

1. Сформувати одновимірний масив цілих чисел, використовуючи генератор випадкових чисел.
2. Роздрукувати отриманий масив.
3. Знищити елементи кратні 7.
4. Додати після кожного непарного елемента масиву елемент зі значенням 0.
5. Роздрукувати отриманий масив.

* Важливі деталі для врахування в імплементації програми:

1. При виконанні роботи використовуються статичні масиви. Для організації статичних масивів із псевдозмінними межами необхідно оголосити масив досить великої довжини
2. При зменшенні або збільшенні довжини масиву необхідно змінювати його реальну довжину.

Завдання №2 **VNS Lab 5**

* Варіант завдання: **24**
* Деталі завдання:

Визначити скільки елементів двовимірного масиву більші від будь-якого елемента на головній діагоналі.

* Важливі деталі для врахування в імплементації програми:

Використовуючи функції, розв’язати зазначене у варіанті завдання. Масив повинен передаватися у функцію як параметр.

Завдання №3 **Algotester Lab 2**

* Варіант завдання: **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:

1 ≤ N ≤ 105

0 ≤ ri ≤ 105

Завдання №4 **Algotester Lab 3**

* Варіант завдання: **2**
* Деталі завдання:

Вам дано два масиви розміром N та M. Значення у цих масивах унікальні. Ваше завдання вивести у першому рядку кількість елемнтів, які наявні в обох масивах одночасно, у другому кількість унікальних елементів в обох масивах разом.

* Важливі деталі для врахування в імплементації програми:

0 ≤ N ≤ 100

0 ≤ ai ≤ 100

0 ≤ M ≤ 100

0 ≤ bi ≤ 100

Завдання №5 **Class Practice**

* Деталі завдання

Реалізувати програму, яка перевіряє, чи дане слово чи число є паліндромом за допомогою рекурсії.Визначте та реалізуйте рекурсивну функцію isPalindrome для рядків.Визначте та реалізуйте перевантажену функцію isPalindrome для цілих чисел. Перетворити ціле число на рядок і використовуйте рядкову версію функції, щоб перевірити, чи це паліндром.

* Важливі деталі для врахування в імплементації програми

Перевантажте функцію *isPalindrome* для роботи з цілими значеннями.*bool isPalindrome*(ціле число);

Завдання №6 **Self Practice**

* Деталі завдання:

<https://algotester.com/en/ContestProblem/DisplayWithEditor/134634>

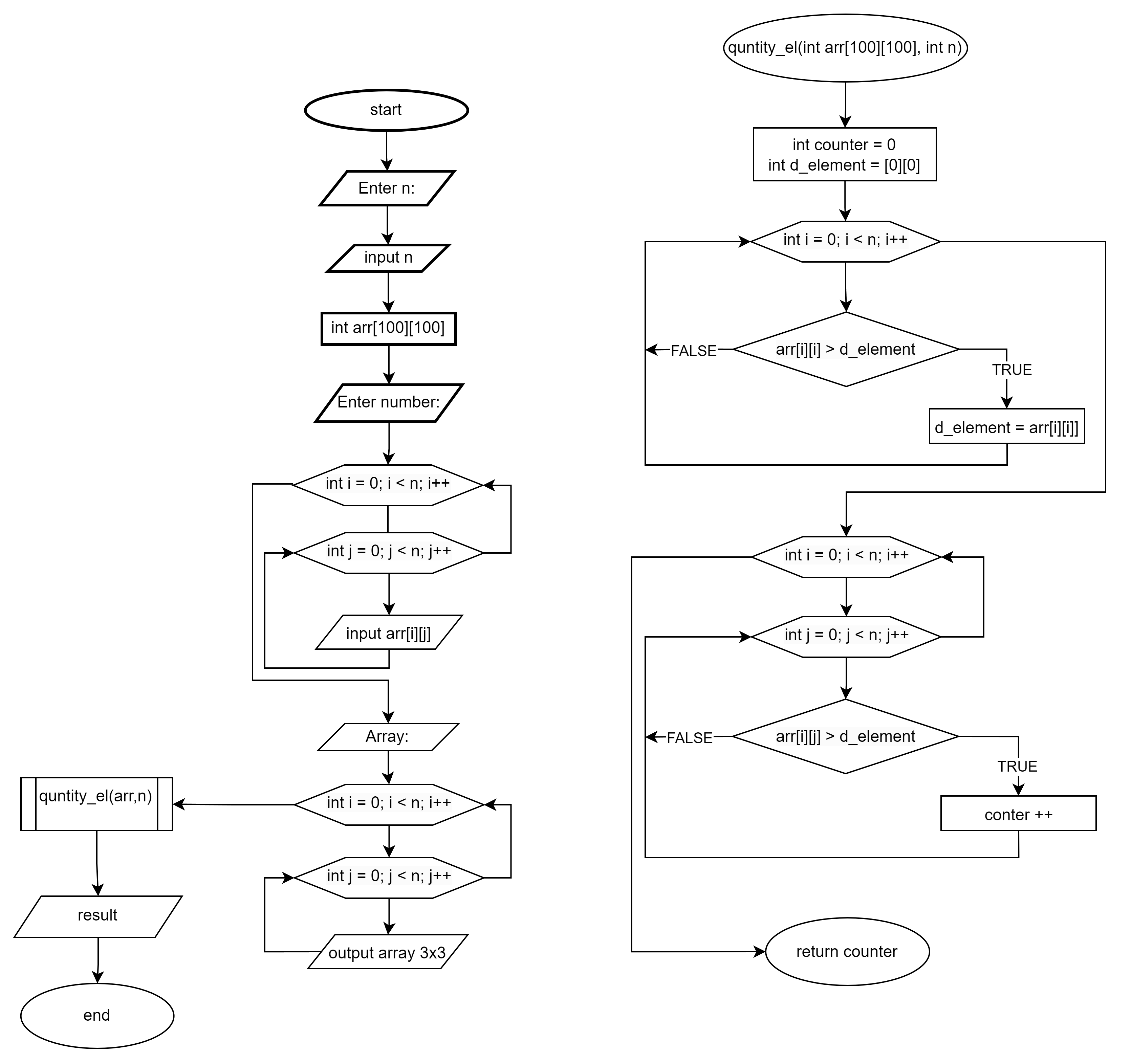
* Важливі деталі для врахування в імплементації програми

−1012≤ai≤1012

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

Програма №2 **VNS Lab 5**

* Блок-схема



**Image 1: Блок-схема до програми**

* Планований час на реалізацію

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

Завдання №1 **VNS Lab 4**

Посилання на файл програми у пул-запиті GitHub

#include <iostream>

#include <cstdlib>

#include <ctime>

using namespace std;

int main() {

    int N = 100;

    int a[N];

    int n;

    cout << "Введіть реальну довжину масиву: ";

    cin >> n;

    cout << "Отриманий масив: ";

    srand(time(NULL));

    for (int i = 0; i < n; i++) {

        a[i] = rand() % 100 + 1;

        cout << a[i] << " ";

    }cout << endl;

    int j = 0;

    for (int i = 0; i < n; i++) {

        if (a[i] % 7 != 0) {

            a[j] = a[i];

            j++;

        }

}

    n=j;

    cout << "Масив після видалення елементів, кратних 7: ";

    for (int i = 0; i < n; i++) {

        cout << a[i] << " ";

    }

cout << endl;

    j = 0;

    for (int i = 0; i < n; i++) {

        if (a[i] % 2 != 0) {

            for (int k = n; k > i; k--) {

                a[k] = a[k - 1];

            }

            a[i + 1] = 0;

            n++;

            i++;

        }

    }

    cout << "Масив після додавання 0 після кожного непарного елемента: ";

    for (int i = 0; i < n; i++) {

        cout << a[i] << " ";

    }

    cout << endl;

    return 0;

}

**Insert 1: Код до програми VNS Lab 4**

Завдання №2 **VNS Lab 5**

Посилання на файл програми у пул-запиті GitHub

#include <iostream>

using namespace std;

int quantity\_el(int arr[100][100], int n) {

int counter = 0;

    int d\_element = arr[0][0];

    for (int i = 0; i < n; i++) {

        if (arr[i][i] > d\_element) {

            d\_element = arr[i][i];

        }

}

    for (int i = 0; i < n; i++) {

        for (int j = 0; j < n; j++) {

            if (arr[i][j] > d\_element) {

                counter++;

            }

        }

}

    return counter;

}

int main() {

    int n;

    cout << "Enter n: ";

    cin >> n;

int arr[100][100];

    cout << "Enter number: ";

    for (int i = 0; i < n; i++) {

        for (int j = 0; j < n; j++) {

            cin >> arr [i][j];

        }

}

    cout << "Array: " << endl;

    for (int i = 0; i < n; i++) {

        for (int j = 0; j < n; j++) {

            cout << arr [i][j] << " ";

        } cout << endl;

}

    int result = quantity\_el(arr, n);

    cout << "Кількість елементів більших від найбільшого елемента на головній діагоналі: " << result <<endl;

    return 0;

}

**Insert 2: Код до програми VNS Lab 5**

Завдання №3 **Algotester Lab 2**

Посилання на файл програми у пул-запиті GitHub

#include <iostream>

#include <vector>

#include <algorithm>

using namespace std;

long find\_tiredness(vector<long> array, long index) {

    array.erase(array.begin() + index);

    return (array[array.size() - 1] - array[0]);

}

int main() {

    long n;

    cin >> n;

    vector<long> r(n);

    for (int i = 0; i < n; i++) {

        cin >> r[i];

}

    sort(r.begin(), r.end());

    int min\_tiredness = r[n-1] - r[0];

    for(int i = 0; i < n; i++) {

        long temporary = find\_tiredness(r, i);

        if (temporary < min\_tiredness) {

            min\_tiredness = temporary;

        }

    }

    cout << min\_tiredness;

    return 0;

}

**Insert 3: Код до програми Algotester Lab 2**

Завдання №4 **Algotester Lab 3**

Посилання на файл програми у пул-запиті GitHub

#include <iostream>

using namespace std;

int main() {

    int n, m;

    cin >> n;

    int a[n];

    for (int i = 0; i < n; i++){

        cin >> a[i];

    }

    cin >> m;

    int b[m];

    for (int i = 0; i < m; i++){

        cin >> b[i];

}

    int counter1 = 0;

    for (int i = 0; i < n; i++) {

        for (int j = 0; j < m; j++) {

            if (a[i] == b[j]) {

                counter1++;

                break;

            }

        }

    }

    cout << counter1 << endl;

    int counter2 = n + m;

    for (int i = 0; i < n; i++) {

        for (int j = 0; j < m; j++) {

            if (a[i] == b[j]) {

                counter2--;

                break;

            }

        }

    }

cout << counter2 << endl;

    return 0;

}

**Insert 4: Код до програми Algotester Lab 3**

Завдання №5 **Class Practice**

Посилання на файл програми у пул-запиті GitHub

#include <iostream>

#include <string>

using namespace std;

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

      if (start >= end) {

        return true;

}

    if (str[start] != str[end]) {

        return false;

    } else {

        return isPalindrome(str, start + 1, end - 1);

    }

}

bool isPalindrome(int number){

string numstr = to\_string(number);

    return isPalindrome(numstr, 0, numstr.length() - 1);

}

int main(){

    string input;

    cout << "Enter a word or number: ";

cin >> input;

bool result = isPalindrome(input, 0, input.length() - 1);

    cout << "This " << (result ? "is" : "isn't") << " palindrom." << endl;

    return 0;

}

**Insert 5: Код до програми Class Practice**

Завдання №6 **Self Practice**

Посилання на файл програми у пул-запиті GitHub

#include <iostream>

using namespace std;

int main() {

    long long sides[5];

    for (int i = 0; i < 5; ++i) {

        cin >> sides[i];

        if (sides[i] <= 0) {

            cout << "ERROR";

            return 0;

        }

        else if (sides[i-1] < sides[i]) {

            cout << "LOSS";

            return 0;

        }

}

    cout << "WIN";

    return 0;

}

**Insert 6: Код до програми Self Practice**

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

Завдання №1 **VNS Lab 4**

INPUT/OUTPUT

Введіть реальну довжину масиву: 13

Отриманий масив: 14 100 77 75 7 34 6 69 53 86 85 59 30

Масив після видалення елементів, кратних 7: 100 75 34 6 69 53 86 85 59 30

Масив після додавання 0 після кожного непарного елемента: 100 75 0 34 6 69 0 53 0 86 85 0 59 0 30

**Insert 7: Результат виконання програми VNS Lab 4**

Програма успішно працює, а саме генерує масив заданого розміру, знищує елементи кратні семи та виводить результуючий масив з додаванням 0 після кожного непарного елемента.

Час затрачений на виконання завдання: 1 год

Завдання №2 **VNS Lab 5**

INPUT/OUTPUT

Enter n: 3

Enter number: 1 2 3 1 5 7 2 0 3

Array:

1 2 3

1 5 7

2 0 3

Кількість елементів більших від будь-якого елемента на головній діагоналі: 1

**Insert 8: Результат виконання програми VNS Lab 5**

Програма успішно працює, а саме повірнює найбільший діагональний елемент з усіма іншими в масиві та виводить кількість елемнтів більших за максимальний діагональний елемент.

Час затрачений на виконання завдання: 1 год

Завдання №3 **Algotester Lab 2**

INPUT/OUTPUT

5

1 2 2 4 4

2

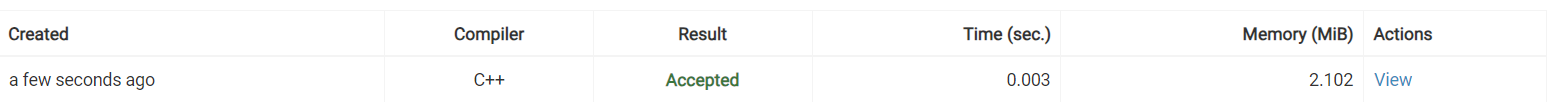
5

1 2 2 2 4

1

**Insert 9: Результат виконання програми Algotester Lab 2**

Програма успішно працює відповідно до всіх вимог поставлених у завданні, а саме мінімізує втому та відповідно пройшла перевірку на алготестері.



**Image 2: Задача зарахована на алготестері**

Час затрачений на виконання завдання: 4 год

Завдання №4 **Algotester Lab 3**

INPUT/OUTPUT

5

1 2 3 4 5

6

3 4 5 6 7 8

3

8

5

1 2 3 4 5

5

4 5 6 7 8

2

8

**Insert 10: Результат виконання програми Algotester Lab 3**

Програма успішно працює відповідно до всіх вимог, а саме виводить кількість спільних елементів у двох масивах та кількість унікальних елементів які наявні в обох масивах. Відповідно задача пройшла перевірку на алготестері.



**Image 3: Задача зарахована на алготестері**

Час затрачений на виконання завдання: 1 год

Завдання №5 **Class Practice**

INPUT/ OUTPUT

Enter a word or number: 123321

This is palindrom.

Enter a word or number: level

This is palindrom.

Enter a word or number: sunny

This is not palindrom

**Insert 11: Результат виконання програми Class Practice**

Програма успішно працює відповідно до всіх вимог поставлених у завданні, а саме правильно визначає чи є введене слов або число паліндромом.

Час затрачений на виконання завдання: 1 год

Завдання №6 **Self Practice**

INPUT/OUTPUT

10 9 8 7 6

WIN

-5 0 4 3 2

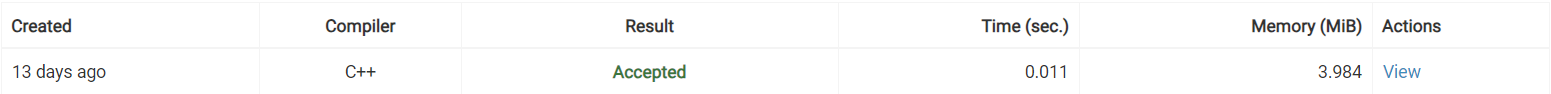
ERROR

1 1 1 1 2

LOSS

**Insert 12: Результат виконання програми Self Practice**

Програма успішно працює відповідно до всіх вимог поставлених у завданні та відповідно пройшла перевірку на алготестері.

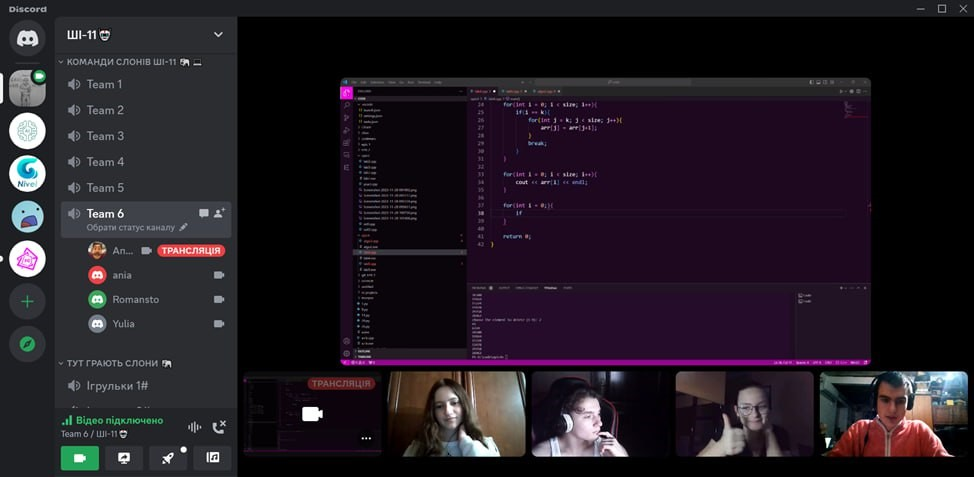


**Image 4: Задача зарахована на алготестері**

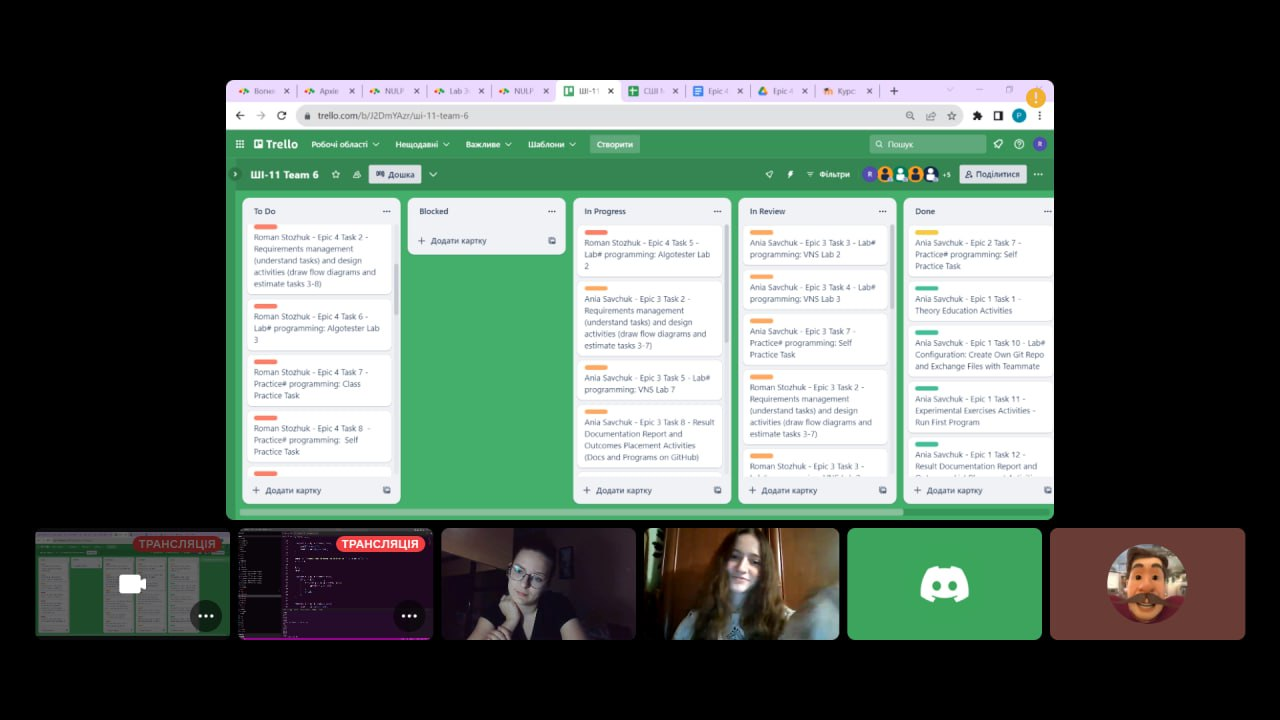
Час затрачений на виконання завдання: 40 хв

## **6. Кооперація з командою:**

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



**Image 5: Скрін зустрічі по обговоренню Епіку 4**



**Image 6: Скрін прогресу в Trello**

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

# **Висновки:**

Опрацювавши теоретичний матеріал по темі, виконала практичні та лабораторні завдання, застосувавши отримані знання. При виконанні поставлених завдань використала одновимірні та двовимірні масиви, тим сами засвоївши теоретичний матеріал на практиці.