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»

***Виконала:***

студентка групи ШІ-12

Борщ Анастасія

# **Тема роботи:**

Ознайомлення з роботою з одновимірними масивами на мові програмування C++. Принципи обробки двовимірних масивів даних. Знати типові алгоритми обробки масивів.

# **Мета роботи:**

Навчитися використовувати одно- та двовимірні масиви в C++. Вивчити основні алгоритми обробки масивів даних. Застосувати вивчене на практиці.

# **Теоретичні відомості:**

1. Теоретичні відомості з переліком важливих тем:

* Тема №1: Одновимірні масиви даних.
* Тема №2: Поняття двовимірний масив.
* Тема №3: Стандартні операції обробки масивів.
* Тема №4: Базові алгоритми сортування масивів.

1. Індивідуальний план опрацювання теорії:

* Тема №1: Одновимірні масиви даних.
  + Джерела Інформації
    - Стаття: <https://www.bestprog.net/uk/2017/03/01/%D0%BC%D0%B0%D1%81%D0%B8%D0%B2%D0%B8-%D1%87%D0%B0%D1%81%D1%82%D0%B8%D0%BD%D0%B0-1-%D0%B2%D0%B8%D0%B7%D0%BD%D0%B0%D1%87%D0%B5%D0%BD%D0%BD%D1%8F-%D0%BC%D0%B0%D1%81%D0%B8%D0%B2%D1%83-%D0%BE%D0%B4/>
  + Що опрацьовано:
    - Опрацьовано принцип роботи з одновимірними масивами, їх специфікацію.
  + Статус: Ознайомлена
  + Початок опрацювання теми: 28.11.2023
  + Звершення опрацювання теми: 28.11.2023
* Тема №2: Поняття двовимірний масив.
  + Джерела Інформації:
    - Стаття: <https://acode.com.ua/urok-81-bagatovymirni-masyvy/>
  + Що опрацьовано:
    - Опрацьовано приклади практичного застосування двовимірних масивів даних в програмному коді
  + Статус: Ознайомлена
  + Початок опрацювання теми: 28.11.2023
  + Звершення опрацювання теми: 28.11.2023
* Тема №3: Стандартні операції обробки масивів.
  + Джерела Інформації:
    - Стаття: <https://cherto4ka.xyz/2020/01/21/%D0%B1%D0%B0%D0%B7%D0%BE%D0%B2%D1%96-%D0%BE%D0%BF%D0%B5%D1%80%D0%B0%D1%86%D1%96%D1%97-%D0%BE%D0%B1%D1%80%D0%BE%D0%B1%D0%BA%D0%B8-%D0%BE%D0%B4%D0%BD%D0%BE%D0%B2%D0%B8%D0%BC%D1%96%D1%80%D0%BD%D0%B8/>
  + Що опрацьовано:
    - Опрацьовано та вивчено основні операції, що спрямовані на обробку масивів
  + Статус: Ознайомлена
  + Початок опрацювання теми: 28.11.2023
  + Звершення опрацювання теми: 28.11.2023
* Тема №4: Базові алгоритми сортування масивів.
  + Джерела Інформації:
    - Відео: <https://www.youtube.com/watch?v=Y5jLCIa3rzM>
    - Відео: <https://www.youtube.com/watch?v=7C15hWlmdyE>
  + Що опрацьовано:
    - Опрацьовано найпоширеніші алгоритми сортування одновимірних масивів
  + Статус: Ознайомлена
  + Початок опрацювання теми: 30.11.2023
  + Звершення опрацювання теми: 30.11.2023

# **Виконання роботи:**

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

Завдання №1 VNS Lab 4

* Варіант завдання: 18
* Деталі завдання : реалізувати за допомогою статичного масиву однонаправлене кільце та виконати кілька дій над його елементами

Завдання №2 VNS Lab 5

* Варіант завдання: 18
* Деталі завдання: Використовуючи функції , масив повинен передаватися у функцію як параметр.

Завдання №3 Algotester Lab 2

* Варіант завдання: 1
* Деталі завдання: <https://algotester.com/uk/ContestProblem/DisplayWithEditor/134635>

Завдання №4 Algotester Lab 3

* Варіант завдання: 3
* Деталі завдання: <https://algotester.com/uk/ContestProblem/DisplayWithEditor/134639>
* Важливі деталі для врахування в імплементації програми

Використання стрічок та циклів для перевірки повторюваності елементів

Завдання №5 Class Practice Work

* Варіант завдання: немає
* Деталі завдання: Написати програму для перевірки стрічки та числа на паліндром

Завдання №6 Self Practice Work

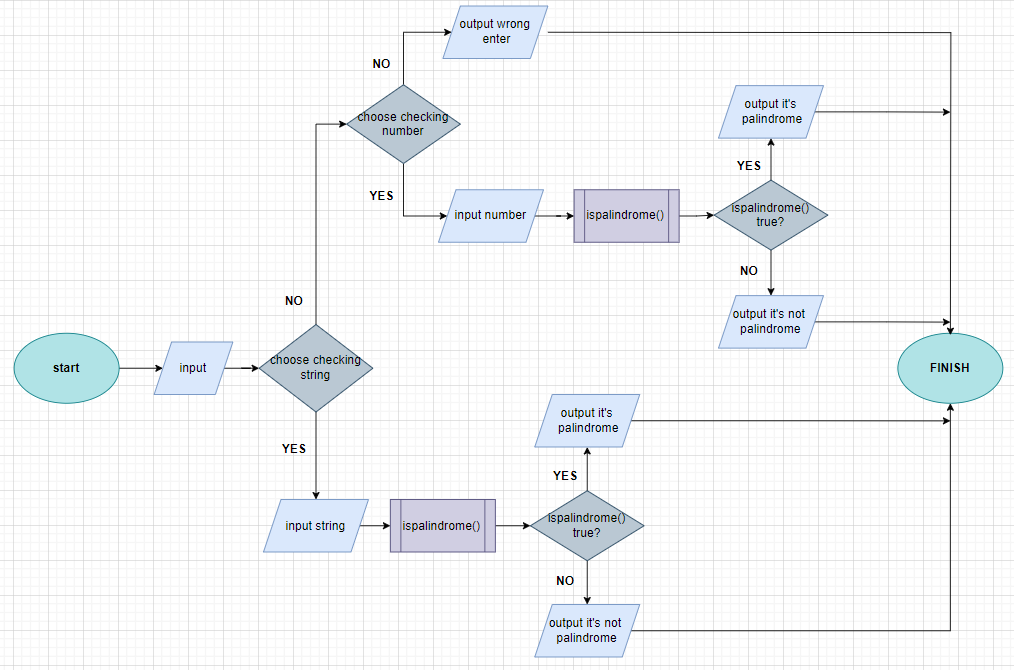
* Варіант завдання: немає
* Деталі завдання: [Коля, Вася і Теніс | Архів | Алготестер](https://algotester.com/uk/ArchiveProblem/DisplayWithEditor/13)
* Важливі деталі для врахування в імплементації програми

Використання циклів for та масиву для обчислення рахунку гри

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

Програма №5 Class Practice Work

* Блок-схема



Output “wro input”

*Рисунок 1. Блок-схема до програми №5*

* Планований час на реалізацію: 60 хвилин

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

Для виконання поставлених завдань додаткова конфігурація середовища не є необхідною

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

Завдання №1 VNS Lab 4

#include <iostream>

#include <vector>

using namespace std;

void print(const vector<int>& arr)

{

    for (int num : arr)

    {

        cout << num << " ";

    }

}

void random(vector<int>& arr, int size)

{

    for (int i = 0; i < size; i++)

    {

        arr.push\_back(rand() % 100);

    }

}

void del(vector<int>& arr, int k)

{

    arr.erase(arr.begin() + k - 1);

}

void pluszero(vector<int>& arr)

{

    int originalSize = arr.size();

    for (int i = 0; i < originalSize; ++i)

    {

        if (arr[i] % 2 == 0)

        {

            arr.insert(arr.begin() + i + 1, 0);

        }

    }

}

int main()

{

    int size;

    cout << "Enter size:" << endl;

    cin >> size;

    vector<int> arr;

    cout << "Here is random array:" << endl;

    random(arr, size);

    print(arr);

    int k;

    cout << endl << "Enter K:" << endl;

    cin >> k;

    del(arr, k - 1);

    cout << "Here is array without k-element:" << endl;

    print(arr);

    cout << endl << "Array when added zeros after each even element:" << endl;

    pluszero(arr);

    print(arr);

    return 0;

}

Завдання №2 VNS Lab 5

#include <iostream>

#include <vector>

using namespace std;

void moveMaxToDiagonal(vector<vector<int>>& matrix) {

    int n = matrix.size();

    for (int i = n - 1; i > 0; --i) {

        int maxVal = matrix[i][i];

        int maxRow = i;

        int maxCol = i;

        for (int j = 0; j < i; ++j) {

            for (int k = 0; k < i; ++k) {

                if (matrix[j][k] > maxVal) {

                    maxVal = matrix[j][k];

                    maxRow = j;

                    maxCol = k;

                }

            }

        }

        swap(matrix[i][i], matrix[maxRow][i]);

        swap(matrix[i][i], matrix[i][maxCol]);

    }

}

bool hasDescendingDiagonal(vector<vector<int>>& matrix) {

    int n = matrix.size();

    for (int i = 0; i < n - 1; ++i) {

        if (matrix[i][i] < matrix[i + 1][i + 1]) {

            return false;

        }

    }

    return true;

}

int main() {

    int n;

    cout << "Enter n: ";

    cin >> n;

    vector<vector<int>> matrix(n, vector<int>(n));

    for (int i = 0; i < n; ++i) {

        for (int j = 0; j < n; ++j) {

            cout << "Enter value of element[" << i + 1 << "][" << j + 1 << "]: ";

            cin >> matrix[i][j];

        }

    }

    cout << "Your Array:" << endl;

    for (int i = 0; i < n; ++i) {

        for (int j = 0; j < n; ++j) {

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

        }

        cout << endl;

    }

    moveMaxToDiagonal(matrix);

    cout << "Modified Array:" << endl;

    for (int i = 0; i < n; ++i) {

        for (int j = 0; j < n; ++j) {

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

        }

        cout << endl;

    }

    if (hasDescendingDiagonal(matrix)) {

        cout << "Sequence is descending." << endl;

    } else {

        cout << "Sequence is not descending." << endl;

    }

    return 0;

}

Завдання №3 Algotester Lab 2

#include <iostream>

#include <vector>

#include <algorithm>

using namespace std;

int main() {

    int N;

    cin >> N;

    vector<int> r(N);

    for (int i = 0; i < N; ++i) {

        cin >> r[i];

    }

    sort(r.begin(), r.end());

    int minFatigue;

    if (N == 1) {

        minFatigue = 0;

    } else if (N == 2) {

        minFatigue = r[1] - r[0];

    } else {

        int option1 = r[N - 2] - r[0];

        int option2 = r[N - 1] - r[1];

        minFatigue = min(option1, option2);

    }

    cout << minFatigue;

    return 0;

}

Завдання №4 Algotester Lab 3

#include <iostream>

#include <string>

using namespace std;

int main() {

    string input, result;

    getline(cin, input);

    char prevChar = input[0];

    int count = 1;

    for (int i = 1; i <= input.size(); ++i) {

        if (input[i] == prevChar) {

            ++count;

        } else {

            result += prevChar;

            if (count > 1) {

                result += to\_string(count);

            }

            prevChar = input[i];

            count = 1;

        }

    }

    cout << result;

    return 0;

}

Завдання №5 Class Practice Work

#include <iostream>

#include <string>

using namespace std;

bool isPalindrome(const string& str, const size\_t start, const size\_t end) {

    if (start >= end) return true;

    return str[start] == str[end] && isPalindrome(str, start + 1, end - 1);

}

bool isPalindrome(const string& str) {

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

}

bool isPalindrome(const long long& num) {

    return isPalindrome(to\_string(num));

}

int main() {

    string str;

    cout << "Enter a string: ";

    cin >> str;

    cout << str << " is " << (isPalindrome(str) ? "" : "not ") << "a palindrome." << endl;

    long long num;

    cout << "Enter an integer: ";

    cin >> num;

    cout << num << " is " << (isPalindrome(num) ? "" : "not ") << "a palindrome." << endl;

    return 0;

}

Завдання №6 Self Practice Work

#include <iostream>

int main() {

    int n, K = 0, V = 0, s[] = {0, 0};

    std::cin >> n;

    char c[n];

    for(int i = 0; i < n; i++) {

        std::cin >> c[i];

        if(c[i] == 'K') {

            s[0]++;

        } else {

            s[1]++;

        }

        if ((s[0] >= 11 || s[1] >= 11) && std::abs(s[0] - s[1]) >= 2) {

            if (s[0] > s[1]) {

                K++;

            } else {

                V++;

            }

            s[0] = s[1] = 0;

        }

    }

    std::cout << K << ":" << V << std::endl;

    if (s[0] > 0 || s[1] > 0) {

        std::cout << s[0] << ":" << s[1];

    }

    return 0;

}

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

Завдання №1 VNS Lab 4

Enter size:

5

Here is random array:

41 67 34 0 69

Enter K:

2

Here is array without k-element:

67 34 0 69

Array when added zeros after each even element:

67 34 0 0 0 0 69

Час затрачений на виконання завдання: 3 години

Завдання №2 VNS Lab 5

Enter n: 5

Enter value of element[1][1]: 3

Enter value of element[1][2]: 4

Enter value of element[1][3]: 6

Enter value of element[1][4]: 5

Enter value of element[1][5]: 4

Enter value of element[2][1]: 3

Enter value of element[2][2]: 9

Enter value of element[2][3]: 8

Enter value of element[2][4]: 7

Enter value of element[2][5]: 0

Enter value of element[3][1]: 2

Enter value of element[3][2]: 1

Enter value of element[3][3]: 6

Enter value of element[3][4]: 9

Enter value of element[3][5]: 3

Enter value of element[4][1]: 7

Enter value of element[4][2]: 5

Enter value of element[4][3]: 0

Enter value of element[4][4]: 8

Enter value of element[4][5]: 6

Enter value of element[5][1]: 5

Enter value of element[5][2]: 4

Enter value of element[5][4]: 2

Enter value of element[5][5]: 2

Your Array:

3 4 6 5 4

3 9 8 7 0

2 1 6 9 3

7 5 0 8 6

5 4 3 2 2

Modified Array:

3 4 6 5 4

3 9 6 8 2

2 8 1 9 3

7 7 0 5 6

5 0 3 2 4

Sequence is not descending.

Час затрачений на виконання завдання : 2 години

Завдання №3 Algotester Lab 2

5

1 2 2 4 4

OUTPUT:

2

Час затрачений на виконання завдання: 3 години



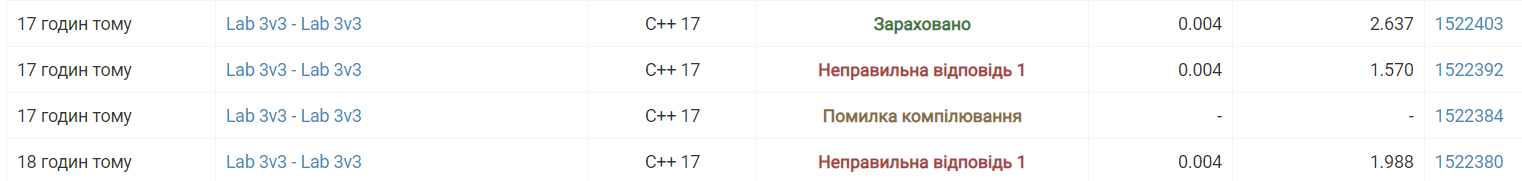
*Рисунок 2. Результат зарахування програми №3 на Алготестер*

Завдання №4 Algotester Lab 3

AAAANNNttyuiooooooo

A4N3t2yuio7

Час затрачений на виконання завдання: 1 година 40 хвилин



*Рисунок 3. Результат зарахування програми №4 на Алготестер*

Завдання №5 Class Practice Work

Enter a string: radar

radar is a palindrome.

Enter an integer: 12

12 is not a palindrome.

Час затрачений на виконання завдання: 2 години

Завдання №6 Self Practice Work

20

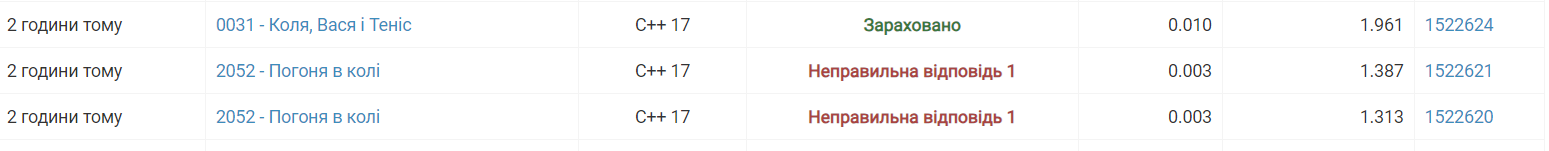
KKVVVVVVVVVVVVVVVKK

VVVKKKVVVVVVVVVVVKK

0:1

2:5

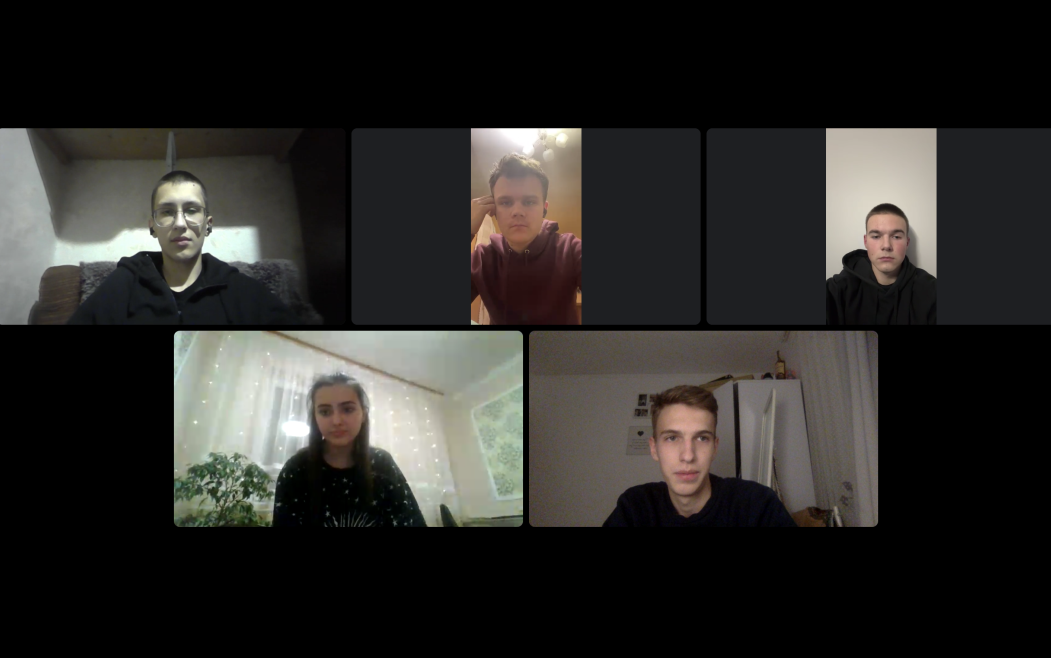
Час затрачений на виконання завдання: 70 хвилин



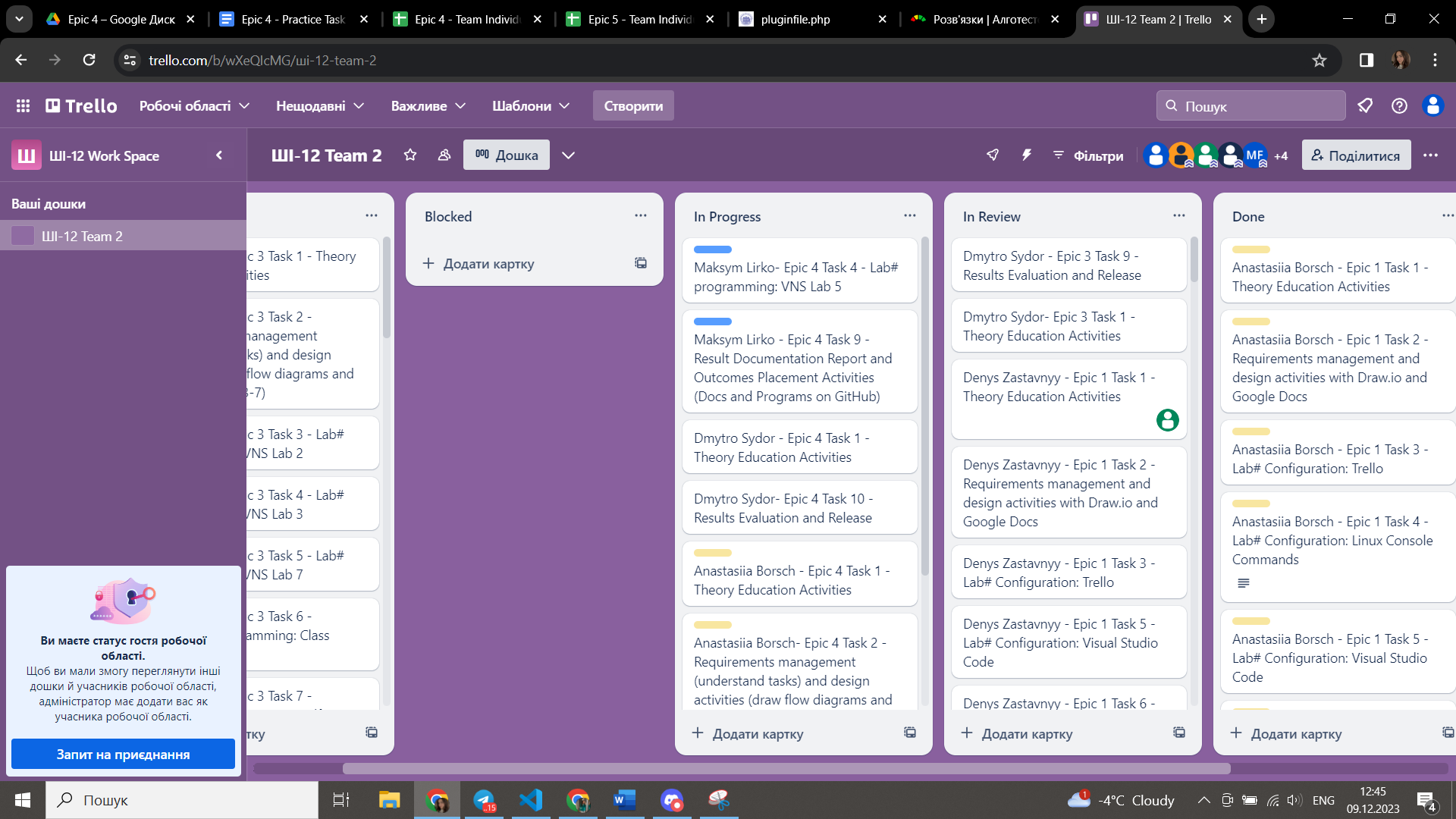
*Рисунок 4. Результат зарахування програми №6 на Алготестер*

## **6. Кооперація з командою:**

* Скрін з зустрічі по обговоренню задач Епіку та Скрін прогресу по Трелло



*Рисунок 5. Скріншот зустрічі*



*Рисунок 6. Скріншот Trello*

# **Висновки:**

Працюючи над теоретичним матеріалом та виконуючи завдання розділу , я навчилась керувати масивами в С++.Під час вивчення матеріалу не виникло суттєвих труднощів. Всі розглянуті матеріали практично гарантовані завдяки їх виконанню в лабораторних і практичних завданнях , а також самостійній практиці. Також був створений базовий алгоритм сортування одновимірних масивів, який необхідний і буде корисний при виконанні завдань в майбутньому.