Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра систем штучного інтелекту



**Звіт**

**про виконання лабораторних та практичних робіт блоку № 4**

На тему:  «Одновимірні масиви. Двовимірні Масиви. Вказівники та Посилання. Динамічні масиви. Структури даних. Вкладені структури. Алгоритми обробки та робота з масивами та структурами.»

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

до:

ВНС Лабораторної Роботи №4

ВНС Лабораторної Роботи №5

Алготестер Лабораторної Роботи №2

Алготестер Лабораторної Роботи №3

Практичних Робіт №4

***Виконала:***

студентка групи ШІ-12

Лебединська Яна Олександрівна

# ***Тема роботи:***

Застосування простих структур даних, таких як масиви. Вивчення одновимірних та двовимірних масивів, які є основою для зберігання та обробки даних, їх застосування. Алгоритми обробки масивів.

# ***Мета роботи:***

Застосування простих структур даних, таких як масиви, в програмуванні. Вивчення одновимірних та двовимірних масивів як ефективних засобів зберігання та обробки інформації. Розглянути алгоритми обробки цих масивів та практичне застосування отриманих знань у програмуванні.

# ***Теоретичні відомості:***

1. Теоретичні відомості з переліком важливих тем:

* Тема №1: Прості структури даних
* Тема №2: Одновимірні масиви
* Тема №3: Двовимірні масиви
* Тема №4: Вектори

1. Індивідуальний план опрацювання теорії:

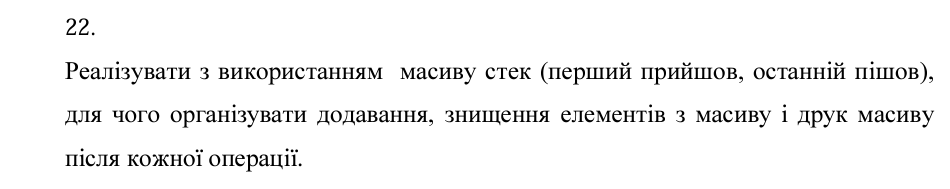
* Тема №1:Прості структури даних
  + Джерела Інформації
    - [chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/http://dspace.wunu.edu.ua/bitstream/316497/24160/1/fkit\_kn\_pzs\_asd\_LEK.pdf](http://dspace.wunu.edu.ua/bitstream/316497/24160/1/fkit_kn_pzs_asd_LEK.pdf) (**книга**)
    - <https://www.google.com/search?q=%D0%BF%D1%80%D0%BE%D1%81%D1%82%D1%96+%D1%81%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D0%B8+%D0%B4%D0%B0%D0%BD%D0%B8%D1%85+%D1%86%D0%B5+%D1%89%D0%BE+%D1%82%D0%B0%D0%BA%D0%B5&sca_esv=586418917&tbm=vid&source=lnms&sa=X&ved=2ahUKEwjyp7OXnOqCAxVyHRAIHXCtDPIQ_AUoAnoECAMQBA&biw=1280&bih=571&dpr=1.5#fpstate=ive&vld=cid:0b6ebfac,vid:Mo8hOFq2-YM,st:0> (**відео**)
  + Що опрацьовано:
    - поняття простих структур даних
  + Статус: Ознайомлена
  + Початок опрацювання теми: 26.11.23
  + Звершення опрацювання теми: 26.11.23
* Тема №2: Одновимірні масиви
  + Джерела Інформації:
    - <https://acode.com.ua/urok-77-masyvy/> (**стаття**)
    - [C++ Теорія ⦁ Урок 40 ⦁ Одновимірні масиви](https://www.youtube.com/watch?v=ULdbOaMBPYc) (**відео**)
  + Що опрацьовано:
    - що таке масив
    - що таке одновимірний масив
    - індекси елементів масиву
    - оголошення масивів фіксованого розміру
  + Статус: Ознайомлена
  + Початок опрацювання теми: 20.10.23
  + Звершення опрацювання теми: 23.10.23
  + По темі зроблено конспект
* Тема №3: Двовимірні масиви
  + Джерела Інформації:
    - <https://acode.com.ua/urok-81-bagatovymirni-masyvy/> (**стаття**)
    - [Двовимірні масиви С++](https://www.youtube.com/watch?v=6dagAMyWTdY) (**відео**)
  + Що опрацьовано:
    - що таке двовимірний масив
    - синтаксис ініціалізації двовимірного масиву
    - застосування таких масивів
  + Статус: Ознайомлена
  + Початок опрацювання теми: 29.10.23
  + Звершення опрацювання теми: 31.10.23
  + По темі зроблено конспект
* Тема №4: Вектори
  + Джерела Інформації:
    - <https://acode.com.ua/urok-99-vvedennya-v-std-vector/> (**стаття**)
  + Що опрацьовано:
    - що таке вектор
    - синтаксис ініціалізації вектора
  + Статус: Ознайомлена
  + Початок опрацювання теми: 25.10.23
  + Звершення опрацювання теми: 26.10.23
  + По темі зроблено конспект.

# ***Виконання роботи:***

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

*Завдання №1*: VNS Lab 4

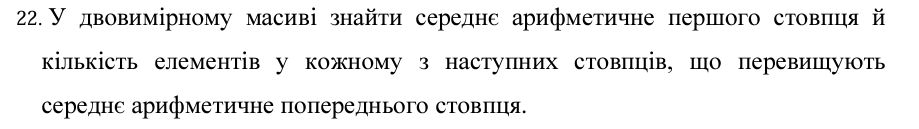
* Варіант №22
* постановка завдання:

**

*умова завдання lab 4 v22*

*Завдання №2*: VNS Lab 5

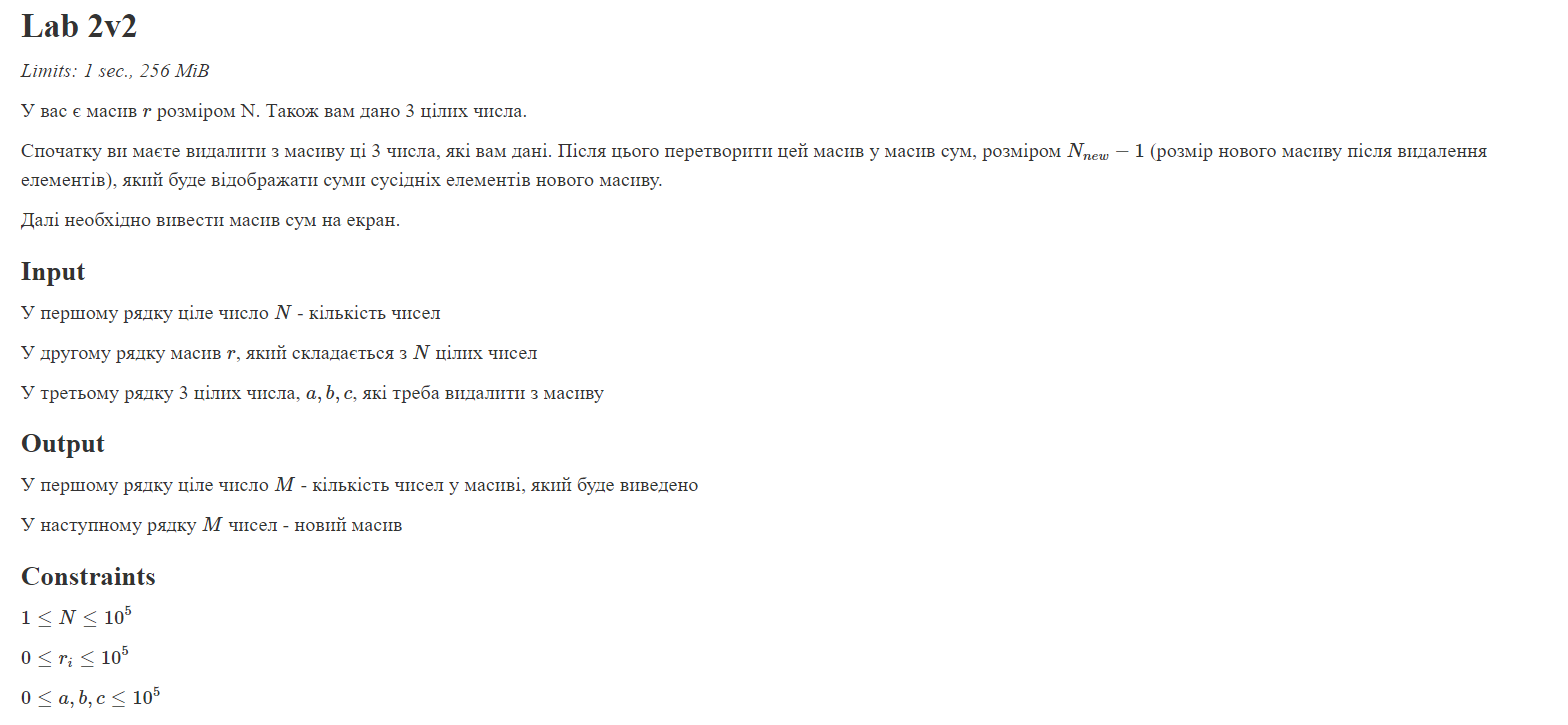
* Варіант №22
* постановка завдання:



*умова завдання lab 5 v22*

*Завдання №3*: Algotester lab 2

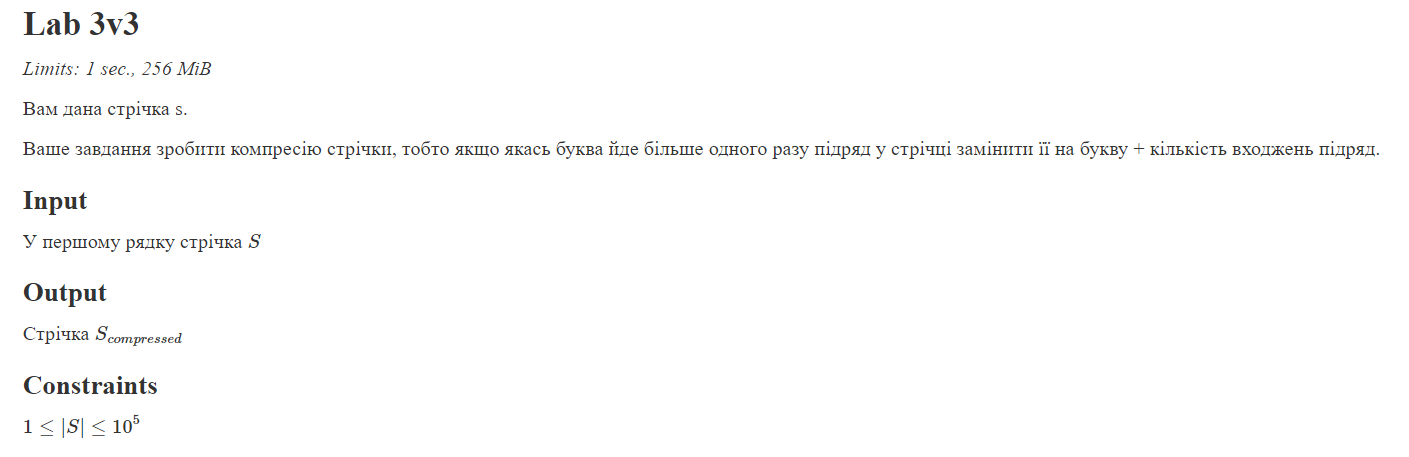
* Варіант №2
* постановка завдання:<https://algotester.com/en/ContestProblem/DisplayWithEditor/134636>



*умова задачі lab 2 v2 Algotester*

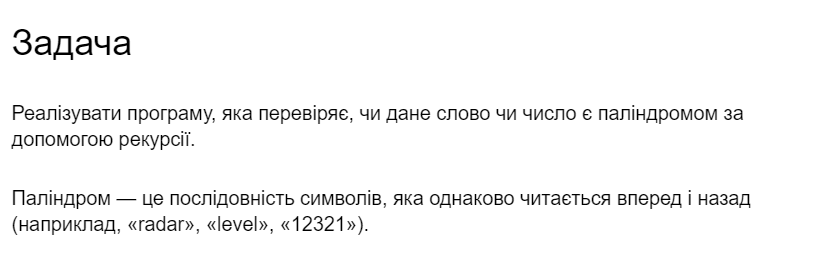
*Завдання №4*: Algotester lab 3

* Варіант №3
* постановка завдання: <https://algotester.com/en/ContestProblem/DisplayWithEditor/134639>



*умова задачі lab 3 v3 Algotester*

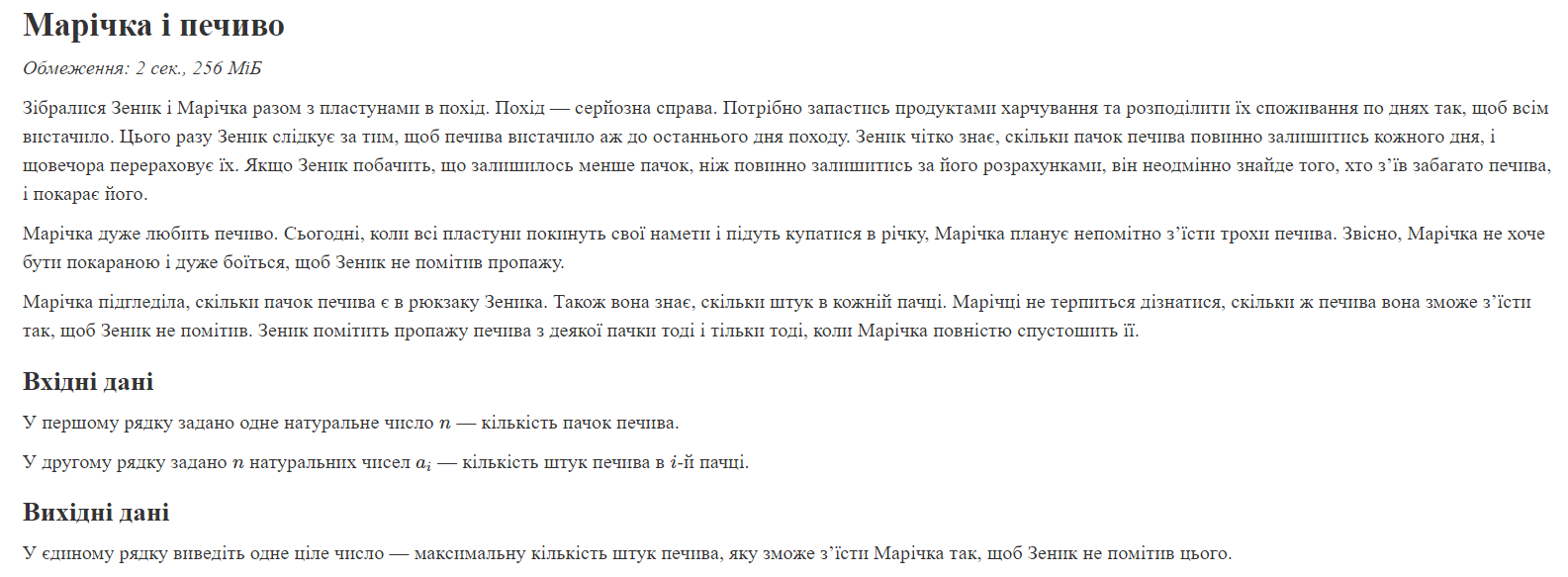
*Завдання №5*: Class Practice Task

* Без варіанту
* постановка завдання: 

*умова задачі class practice*

*Завдання №*6 Self Practice Task (algotester)

* Без варіанту
* Постановка завдання: <https://algotester.com/uk/ArchiveProblem/DisplayWithEditor/2>

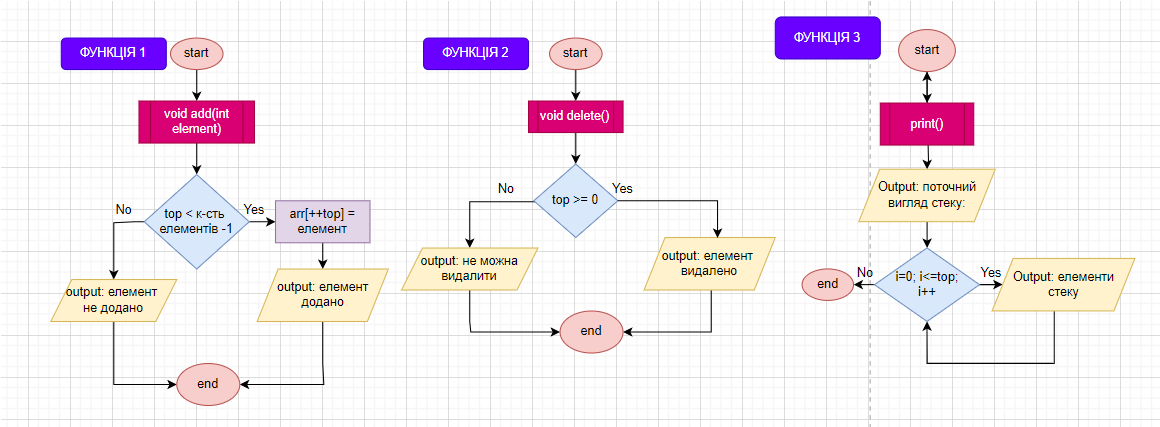


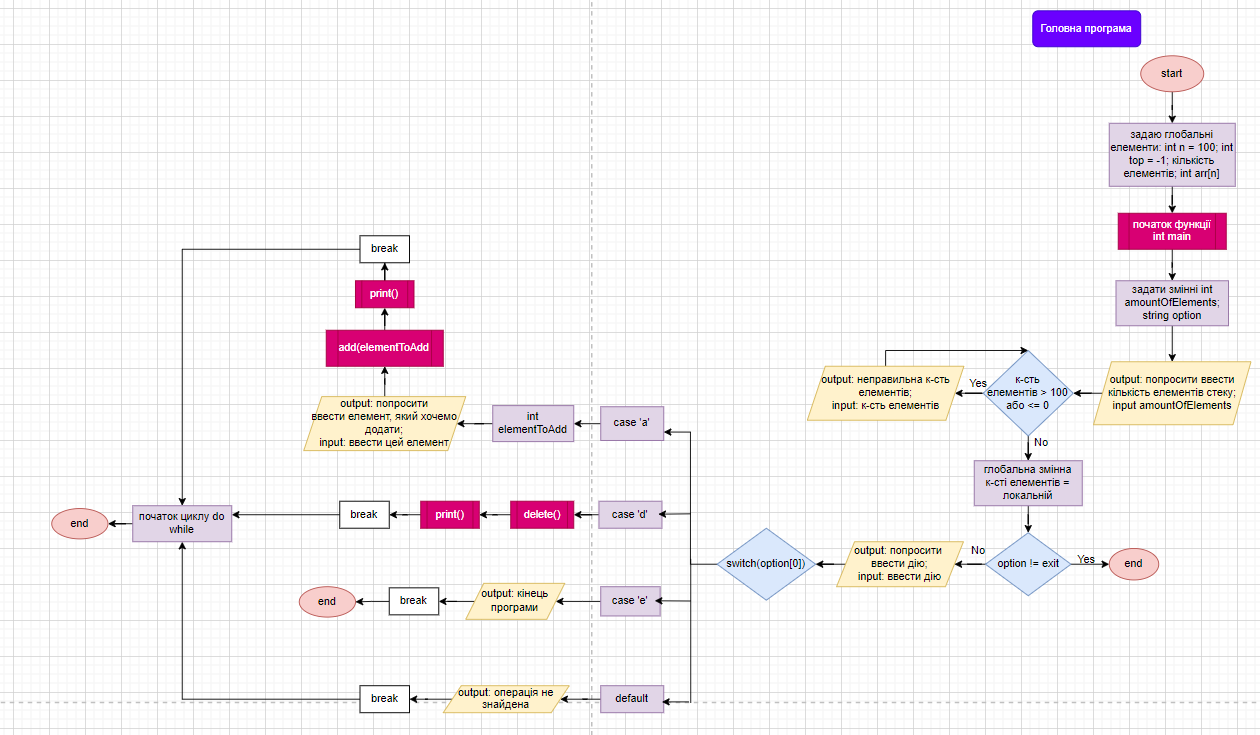
*умова задачі self practice Algotester*

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

***Програма №1*** VNS Lab 4

* Блок-схема





* Планований час на реалізацію: 1 год 20 хв

***Програма №2*** VNS Lab 5

* Планований час на реалізацію 40 хв

***Програма №3*** Algotester lab 2

* Планований час на реалізацію: 1 год

***Програма №4*** Algotester lab 3

* Планований час на реалізацію: 40 хв

***Програма №5*** Class Practice Task

* Планований час на реалізацію: 30-40 хв

***Програма №6*** Self Practice Task (algotester)

Планований час на реалізацію: 30 хв

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

Додаткової конфігурації не потрібно.

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

**Завдання №1** VNS Lab 4

Посилання на файл програми у пул-запиті GitHub: <https://github.com/artificial-intelligence-department/ai_programming_playground/blob/439a51b8c240e21ecf32296cbddae8365935fb81/ai_12/yana_lebedynska/Epic4/vns_lab_4_v22_yana_lebedynska.cpp>

**#include <iostream>**

**using namespace std;**

**const int n = 100;**

**int arr[n];**

**int top = -1;**

**int amountOfELEMENTS;**

**void add(int element)**

**{**

**cout << endl;**

**if(top < amountOfELEMENTS-1)**

**{**

**arr[++top] = element;**

**cout << element << " is successfully added!" << endl;**

**}**

**else**

**{**

**cout << "The element " << element << " can't be added! Stack is overwhelmed." << endl;**

**}**

**}**

**void deleteElement()**

**{**

**cout << endl;**

**if(top >= 0)**

**{**

**cout << arr[top--] << " is now succssesfully deleted!" << endl;**

**}**

**else**

**{**

**cout << "You can't delete the element! Stack is empty." << endl;**

**}**

**}**

**void print()**

**{**

**cout << "Current stack: ";**

**for(int i = 0; i <= top; i++)**

**{**

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

**}**

**cout << endl;**

**cout << endl;**

**}**

**int main()**

**{**

**int amountOfElements;**

**string option;**

**cout << "Enter the amount of elements of the stack: ";**

**cin >> amountOfElements;**

**while(amountOfElements > 100 || amountOfElements <=0)**

**{**

**cout << "The amount of elements is wrong! It must be greater than 0 and less than 100." << endl;**

**cout << "Try again: ";**

**cin >> amountOfElements;**

**}**

**amountOfELEMENTS = amountOfElements;**

**do**

**{**

**cout << "You can:" << endl;**

**cout << "1. Add an element (type add)" << endl;**

**cout << "2. Delete an element (type delete)" << endl;**

**cout << "3. Exit the program (type exit)" << endl;**

**cout << "Enter the option here: ";**

**cin >> option;**

**switch(option[0])**

**{**

**case 'a':**

**int elementToAdd;**

**cout << "Enter the element you want to add: ";**

**cin >> elementToAdd;**

**add(elementToAdd);**

**print();**

**break;**

**case 'd':**

**deleteElement();**

**print();**

**break;**

**case 'e':**

**cout << "The end of the program." << endl;**

**break;**

**default:**

**cout << "The option isn't found! Try again." << endl;**

**break;**

**}**

**}**

**while(option != "exit");**

**return 0;**

**}**

*код задачі lab 4*

**Завдання №2** VNS Lab 5

Посилання на файл програми у пул-запиті GitHub: <https://github.com/artificial-intelligence-department/ai_programming_playground/blob/439a51b8c240e21ecf32296cbddae8365935fb81/ai_12/yana_lebedynska/Epic4/vns_lab_5_v22_yana_lebedynska.cpp>

#include <iostream>

using namespace std;

int main()

{

int arr[4][5] =

{

{1, 4, 5, 2, 7},

{2, 4, 9, 0, 3},

{3, 4, 1, 1, 2},

{3, 1, 6, 8, 2}

};

int sumOfFirstColumn = 0;

double avarageOfFirstColumn;

for(int row = 0; row < 4; row++)

{

sumOfFirstColumn += arr[row][0];

}

avarageOfFirstColumn = sumOfFirstColumn/4;

cout << "The avarage of elements of the first column is: " << avarageOfFirstColumn << endl;

for(int col = 1; col < 5; col++)

{

int numberOfElements = 0;

for(int row = 0; row < 4; row++)

{

if(arr[row][col] > avarageOfFirstColumn)

{

numberOfElements++;

}

}

cout << "In column number " << col << " there is " << numberOfElements << " elements which exceed the average. " << endl;

cout << endl;

}

return 0;

}

## *код задачі lab 5*

**Завдання №3** Algotester lab 2 v2

Посилання на файл програми у пул-запиті GitHub: <https://github.com/artificial-intelligence-department/ai_programming_playground/blob/439a51b8c240e21ecf32296cbddae8365935fb81/ai_12/yana_lebedynska/Epic4/algotester_lab_2_yana_lebedynska.cpp>

#include <iostream>

#include <vector>

using namespace std;

int main() {

long N, a, b, c;

cin >> N;

vector<long> r(N);

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

{

cin >> r[i];

}

cin >> a >> b >> c;

vector<long> newVector;

bool elementToDeletePresent = false;

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

{

if (r[i] == a || r[i] == b || r[i] == c)

{

elementToDeletePresent = true;

}

else

{

newVector.push\_back(r[i]);

}

}

long M = newVector.size();

if (M <= 1)

{

cout << 0;

}

else

{

cout << M - 1 << endl;

if (M > 1) {

vector<long> sum;

if (M - 1 > 0)

{

sum.resize(M - 1);

for (int i = 0; i < M - 1; i++)

{

sum[i] = newVector[i] + newVector[i + 1];

}

}

for (int i = 0; i < M - 1; i++)

{

cout << sum[i] << " ";

}

}

}

return 0;

}

## *код задачі algotester lab2 v2*

**Завдання №4** *Algotester lab3 v3*

Посилання на файл програми у пул-запиті GitHub: <https://github.com/artificial-intelligence-department/ai_programming_playground/blob/439a51b8c240e21ecf32296cbddae8365935fb81/ai_12/yana_lebedynska/Epic4/algotester_lab_3_yana_lebedynska.cpp>

#include <iostream>

#include <string>

using namespace std;

int main() {

string S;

cin >> S;

long n = S.size();

string compressedS;

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

int count = 1;

while (i < n - 1 && S[i] == S[i + 1])

{

count++;

i++;

}

compressedS += S[i];

if (count > 1)

{

compressedS += to\_string(count);

}

}

cout << compressedS << endl;

return 0;

}

## *код задачі algotester lab 3 v3*

**Завдання №5** Class Practice Task

Посилання на файл програми у пул-запиті GitHub: <https://github.com/artificial-intelligence-department/ai_programming_playground/blob/439a51b8c240e21ecf32296cbddae8365935fb81/ai_12/yana_lebedynska/Epic4/practice_work_team_task_yana_lebedynska.cpp>

#include <iostream>

#include <string>

using namespace std;

bool isPalindrom(const string& WORD, int start, int end)

{

if(WORD[start] != WORD[end])

{

return false;

}

if(start == end)

{

return true;

}

return isPalindrom(WORD, start + 1, end - 1);

}

bool isPalindrom(int number)

{

string numberSTR = to\_string(number);

return isPalindrom(numberSTR, 0, numberSTR.length() - 1);

}

int main()

{

string LINE;

int NUM;

cout << "Enter the word you want to check: ";

cin >> LINE;

cout << "Enter the number you want to check: ";

cin >> NUM;

if(isPalindrom(LINE, 0, LINE.length() - 1))

{

cout << "Yes, the word " << LINE << " is palidrom!" << endl;

}

else

{

cout << "No, the word " << LINE << " isn't palindrom!" << endl;

}

if(isPalindrom(NUM))

{

cout << "Yes, the number " << NUM << " is palindrom!";

}

else

{

cout << "No, the number " << NUM << " isn't palindrom!";

}

return 0;

}

## *код задачі class practice*

**Завдання №6** Self Practice Task (Algotester)

Посилання на файл програми у пул-запиті GitHub: <https://github.com/artificial-intelligence-department/ai_programming_playground/blob/439a51b8c240e21ecf32296cbddae8365935fb81/ai_12/yana_lebedynska/Epic4/practice_work_self_algotester_tasks_yana_lebedynska.cpp>

#include <iostream>

#include <vector>

using namespace std;

int main()

{

long long packs;

cin >> packs;

std::vector<long long> cookies(packs);

for (long long i = 0; i < packs; i++) {

std::cin >> cookies[i];

}

long long can\_eat = 0;

for (long long i = 0; i < packs; i++)

{

long long max = cookies[i] - 1;

can\_eat += max;

}

cout << can\_eat;

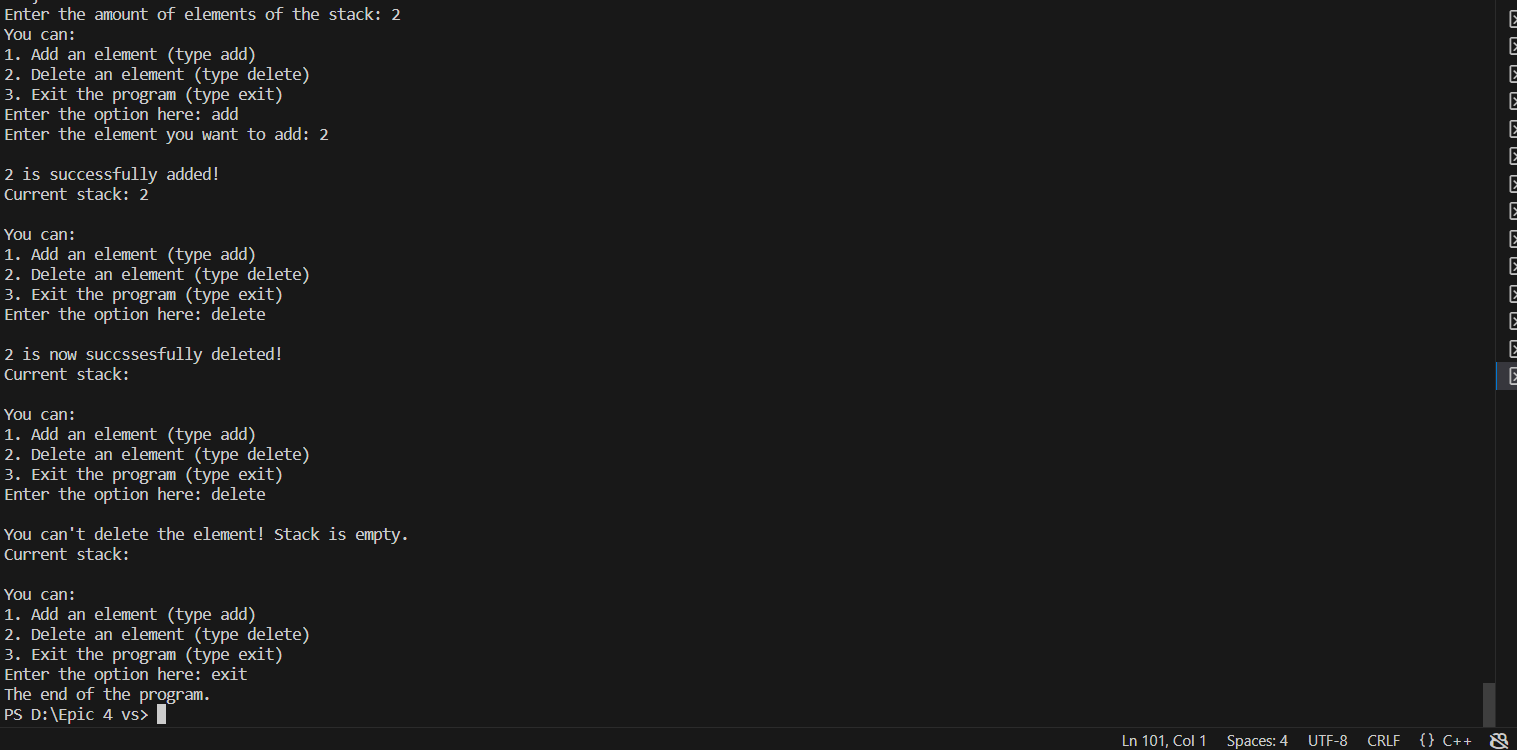
return 0;

}

## *код задачі self practice (algotester)*

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

***Завдання №1*** VNS Lab 4

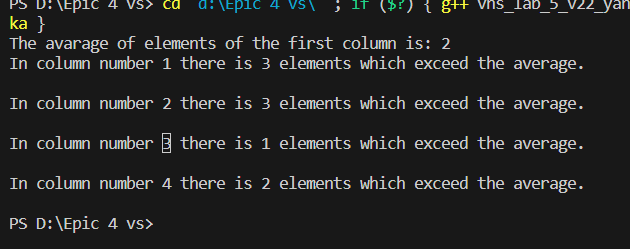
****

*робота програми*

Час затрачений на виконання: 2 год 10 хв

***Завдання №2*** VNS Lab 5

**Output:**



*результат програми*

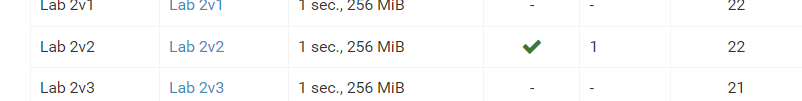
Час затрачений на виконання завдання: 50 хв

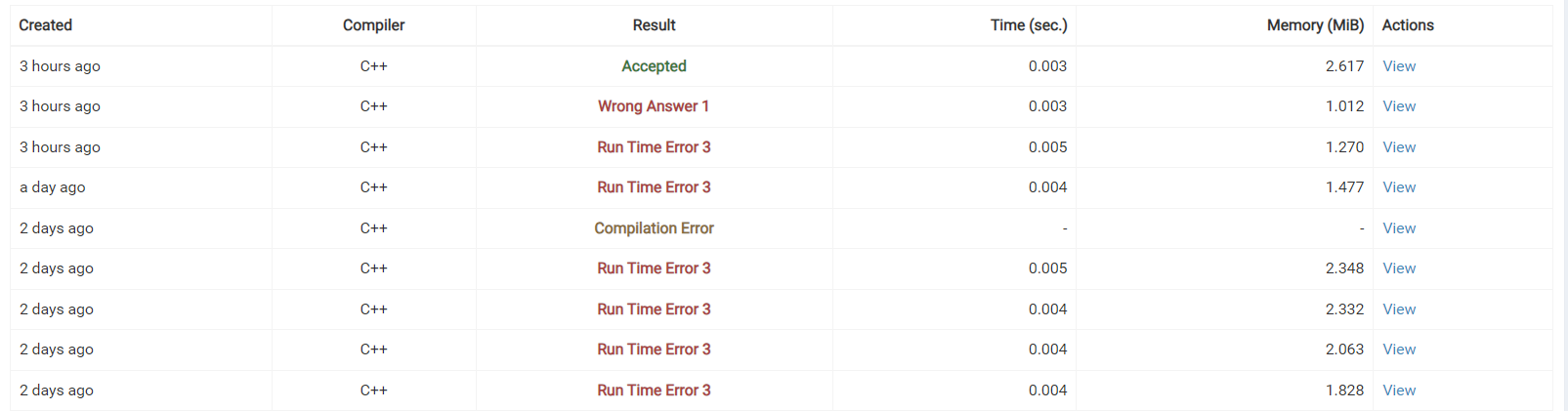
***Завдання №3*** Algotester lab2 v2

**Input:**



**Evidences:**

****

****

Час затрачений на виконання: 1 год 30 хв

***Завдання №4*** Algotester lab 3 v3

**Input:**

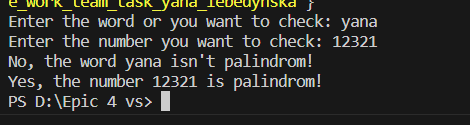
****

**Output:**

**** Час затрачений на виконання: 40 хв

***Завдання №5*** Class Practice Task

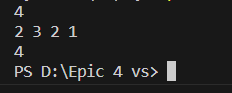
**Результат програми:**

****

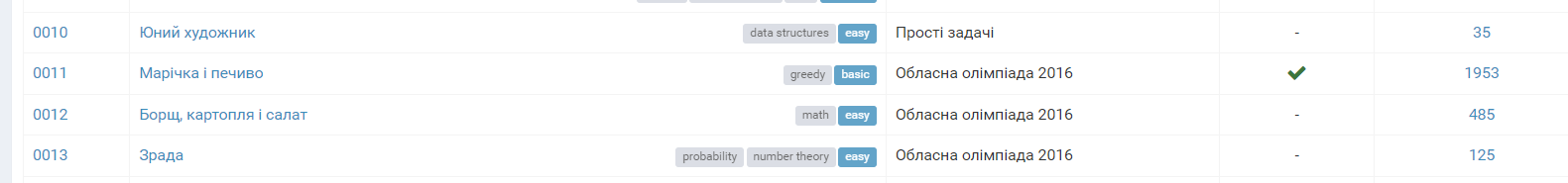
Час затрачений на виконання: 1 год

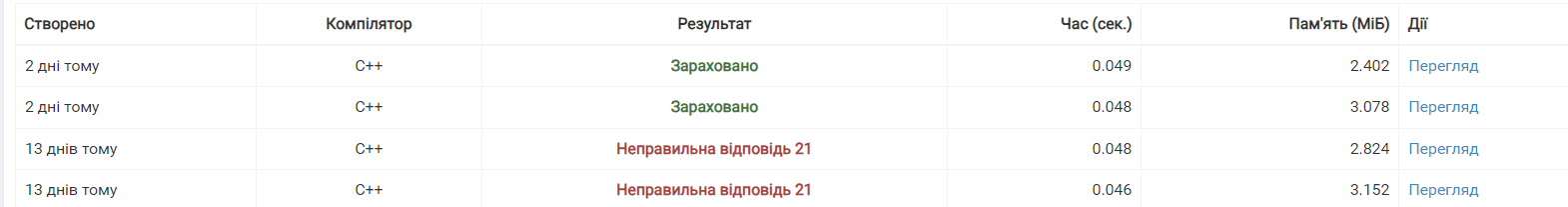
***Завдання №6*** Self Practice Task (Algotester)

**Результат програми:**

****

**Evidences:**

****

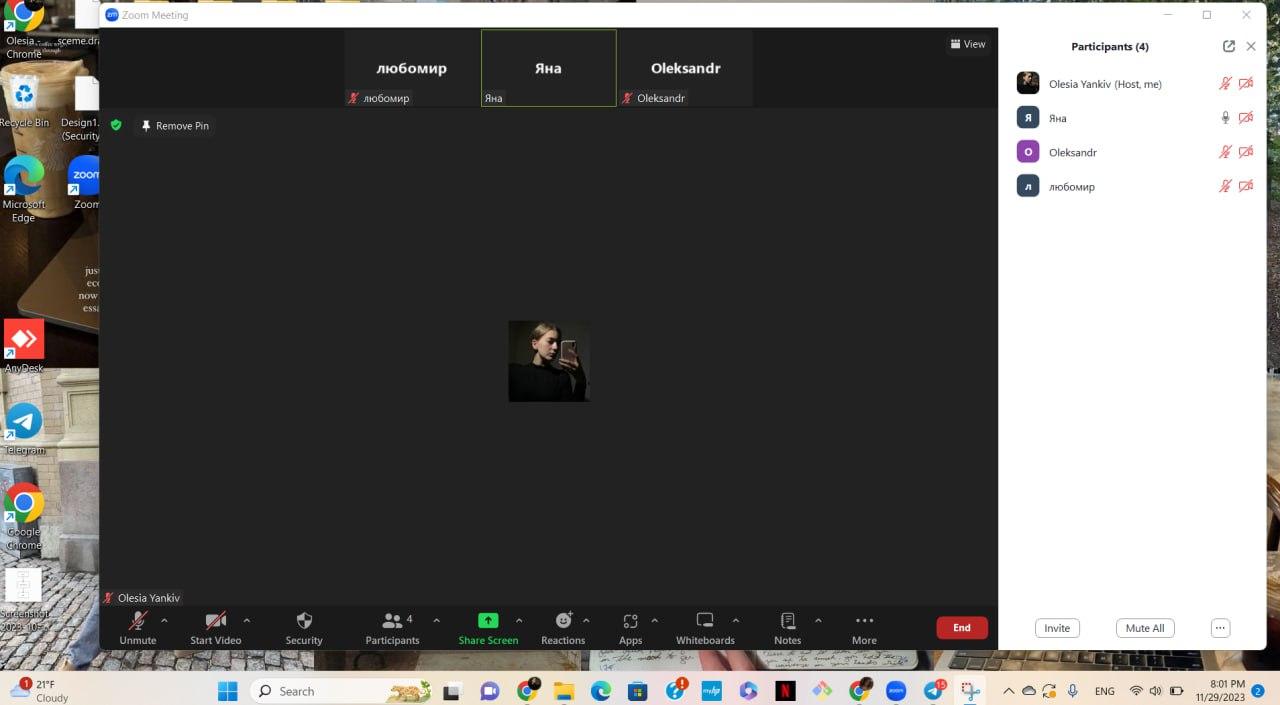
****

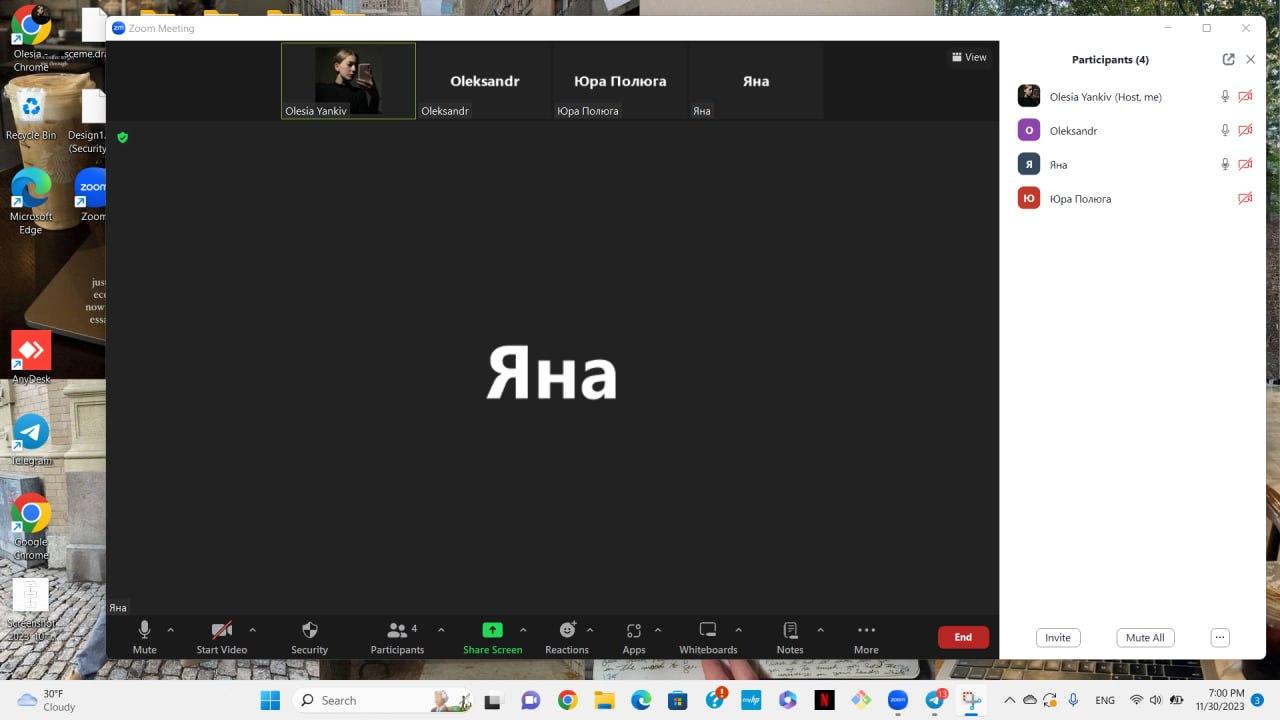
Час затрачений на виконання: 35 хв

# 

# **Робота команди:**

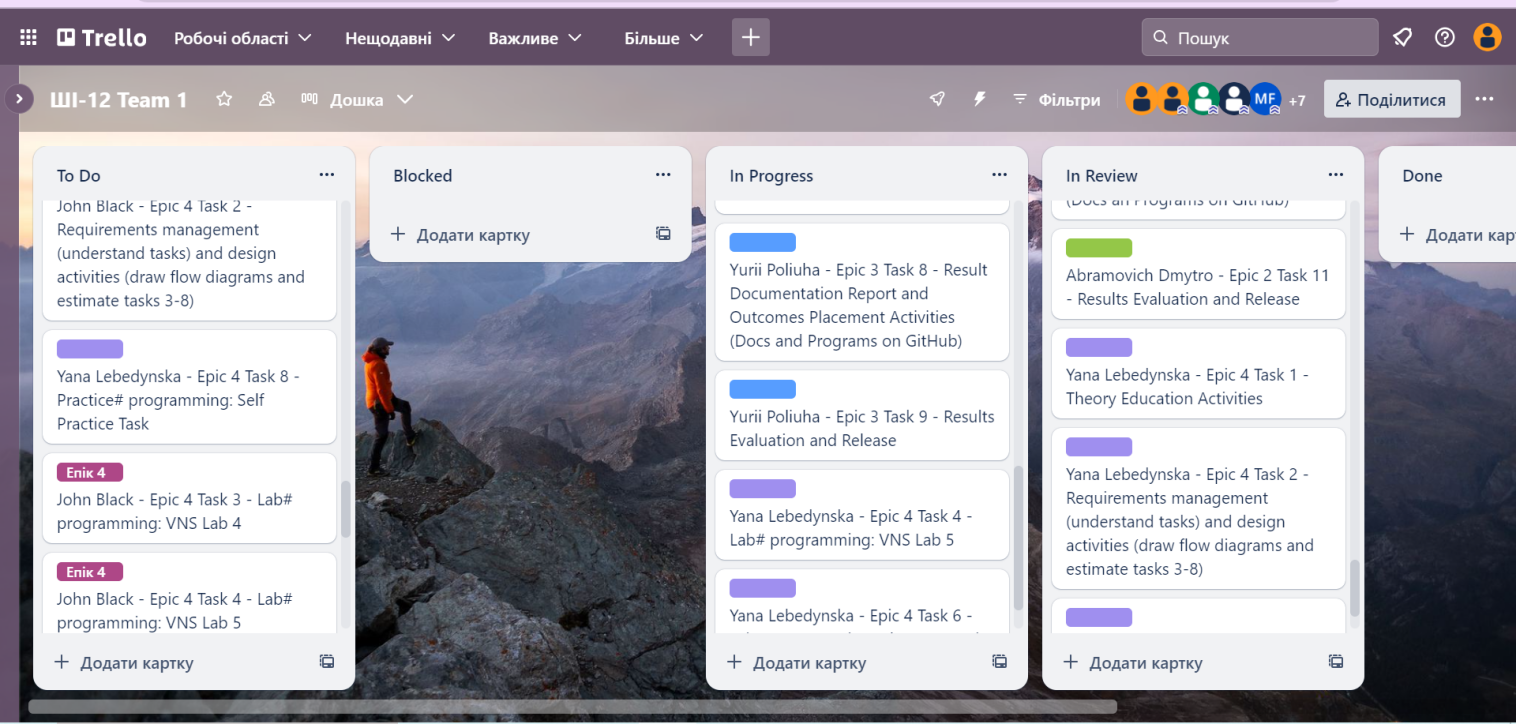
**Міти**

****

****

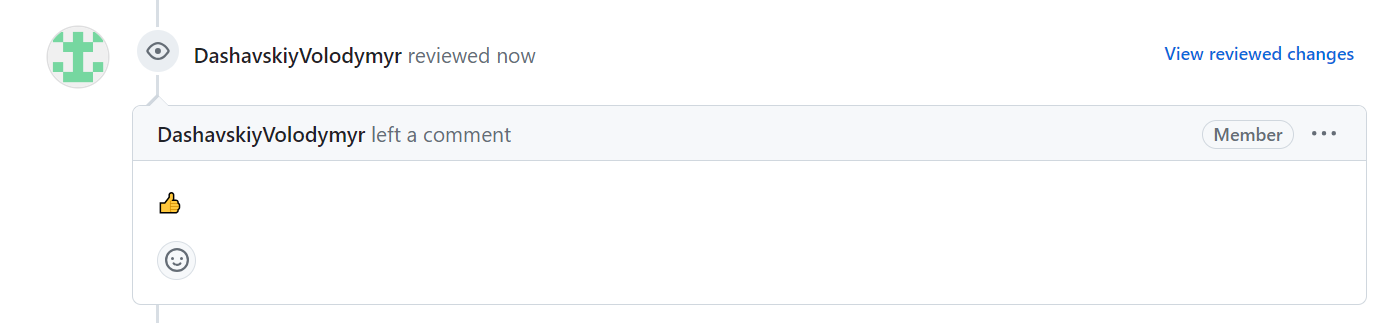
*скріни мітів у зумі*

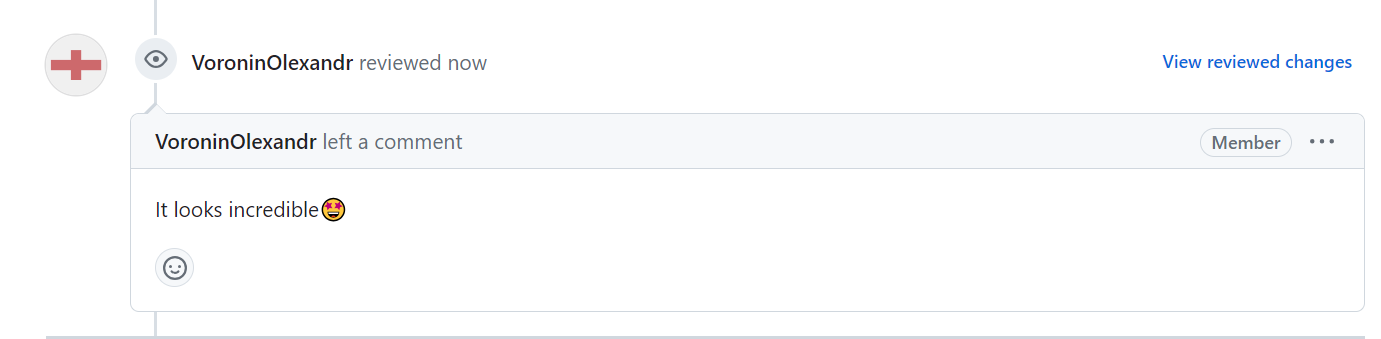
**Трелло**

****

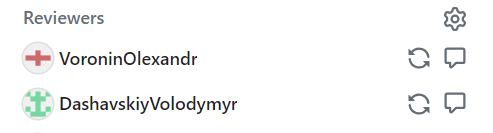
*скрін робочої області в трелло*

**Пул-реквест: <https://github.com/artificial-intelligence-department/ai_programming_playground/pull/588>**

****

****

*коментар до пул-реквесту в гітХабі*

****

**Висновки:**

Протягом епіка 4 я змогла навчитись застосовувати в програмуванні одновимірні та двовимірні масиви, вектори, зрозуміти їх зручність та особливості. Також працювала з функціями та циклами. За допомогою практики закріпила свої новонабуті навики.