## 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***: «Епік 4: Прості структури даних. Одновимірні масиви. Двовимірні Масиви. Алгоритми обробки»

***Виконав:***

студент групи ШІ-13

Гаргай Юрій

Львів 2023

# Тема роботи:

# Робота із простими структурами даних та алгоритмами обробки. Робота з одновимірними та двовимірними масивами.

**Задачі:**

1. *VNS Lab 4*
2. *VNS Lab 5*
3. *Algotester Lab 2*
4. *Algotester Lab 3*
5. *Practice*
6. *Self-practice*

# Мета роботи:

# Ознайомитися з простими струкурами данних

# Ознайомитись з одновимірними та двовимірними масивами

# Ознайомитись з алгоритмами обробки

# Теоретичні відомості:

* 1. Теоретичні відомості з переліком важливих тем:
* Тема №1: Прості структури даних
* Тема №2: Одновимірні та двовимірні масиви
* Тема №3: Алгоритми обробки
  1. Індивідуальний план опрацювання теорії:
* Тема №1: Прості структури даних
* Джерела Інформації:
  + - https://www.geeksforgeeks.org/data-structures/
    - <https://ua.udemy.com/>

Beginning C++ Programming - From Beginner to Beyond

* Що опрацьовано:
* Ознайомився з простими структурами даних та їхніми типами
  + Статус: Ознайомлений
  + Початок опрацювання теми: 28.11.2023
  + Звершення опрацювання теми: 28.11.2023
* Тема №2: Одновимірні та двовимірні масиви
* Джерела Інформації:
  + - <https://ua.udemy.com/>

Beginning C++ Programming - From Beginner to Beyond

* + Що опрацьовано**:**
    - Ознайомився з одновимірними та двовимірними масивами
  + Статус: Ознайомлений
  + Початок опрацювання теми:15.10.2023
  + Звершення опрацювання теми: 17.10.2023

* Тема №3: Алгоритми обробки
  + Джерела Інформації:
    - <https://ua.udemy.com/>

Beginning C++ Programming - From Beginner to Beyond

* + Що опрацьовано:
    - Ознайомився з алгоритмами обробки
  + Статус: Ознайомлений
  + Початок опрацювання теми: 30.11.2023
  + Звершення опрацювання теми: 01.12.2023

# Виконання роботи:

**Завдання №1**: VNS Lab 4

Варіант завдання: **13**

Деталі завдання

1) Сформувати одновимірний масив цілих чисел, використовуючи генератор

випадкових чисел.

2) Роздрукувати отриманий масив.

3) Знищити з масиву всі елементи, які співпадають з його мінімальним

значенням.

4) Додати на початок масиву 3 елементи зі значенням, яке дорівнює середньому

арифметичному масиву.

5) Роздрукувати отриманий масив.

**Планована оцінка часу виконання завдань:** 1 день

**Завдання №2** VNS Lab 5

* Варіант завдання: **13**
* Деталі завдання: Визначити чи можна у двовимірному масиві знайти такий стовпець, що

розбиває масив на два так, що сума елементів у першому більша, ніж сума

елементів у другому. Сам стовпець у розбиті частини не входить.

**Планована оцінка часу виконання завдань:** 1 день

**Завдання №3** Algotester Lab 2

* Варіант завдання: **2**
* Деталі завдання:
* У вас є масив r розміром N. Також вам дано 3 цілих числа.Спочатку ви маєте видалити з масиву ці 3 числа, які вам дані. Після цього перетворити цей масив у масив сум, розміром New-1 нового масиву після видалення елементів), який буде відображати суми сусідніх елементів нового масиву. Далі необхідно вивести масив сум на екран.
* **Планована оцінка часу виконання завдань:** 1 день

**Завдання №4** Algotester Lab 3

* Варіант завдання: **13**
* 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. **Планована оцінка часу виконання завдань:** 1 день

**Завдання №5** Practice

* Деталі завдання:
* Реалізувати програму, яка перевіряє, чи дане слово чи число є паліндромом за допомогою рекурсії.

**Планована оцінка часу виконання завдань:** 1 день

**Завдання №6** Self-practice

Ви створюєте просту програму керування масивом за допомогою функцій.

Програма повинна вміти:

* + Вивести масив на екран.
  + Додати число до масиву.
  + Знайти середє значення масиву
  + Знайти найменше значення масиву
  + Знайти найбільше значення масиву

**Планована оцінка часу виконання завдань:** 1 день

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

Завдання №6Self-practice

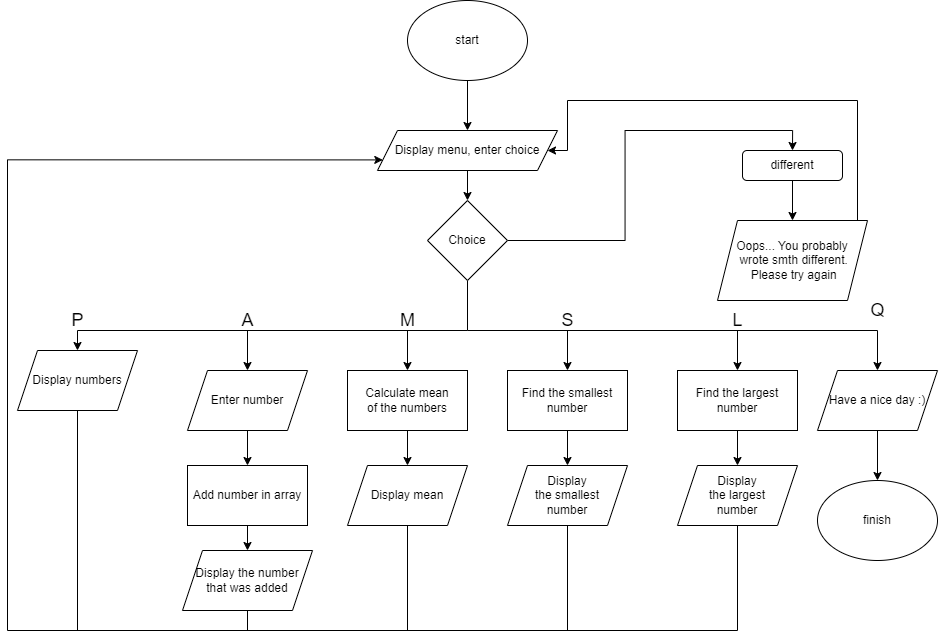
* + Блок-схема

Рисунок 1 блок схема

Планований час на реалізацію: 1 день

sssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssss

sssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssss

### Код програм з результатом та посиланням на зовнішні ресурси:

Завдання №1 VNS Lab 4

1)Код програми

#include <iostream>

#include <cstdlib>

#include <ctime>

#include <iomanip>

using namespace std;

int smallest();

double find\_mean();

const int N {100};

size\_t n {10};

int mini {};

size\_t NewSize {3};

int old[N];

double updated[N];

int main() {

    srand(time(0));

    for(int i = 0; i < n; i++) { //генерація рандомного масиву

        old[i] = rand();

    }

    for(int i = 0; i < n; i++) {

        cout << old[i] << " ";

    }

    cout << endl <<"-------------------------------------------------------------------" << endl;

    cout << fixed << setprecision(1);

    for(int i = 0; i<3; i++){   //3 значення середнього арифметичного на початок

        updated[i] = find\_mean();

    }

    mini = smallest();              // Генерація оновленого масиву без найменшого

    for (int i=0; i < n; i++){

        if ( old[i] != mini){

            updated[NewSize] = old[i];

            NewSize++;

        }

    }

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

        cout << updated[i] << " ";

    }

    return 0;

}

int smallest(){                     // функція найменшого числа

    int minimum = INT\_MAX;

        for (int i= 0; i < 10; i++){

        if (old[i]< minimum)

            minimum = old[i];

        }

    return minimum;

}

double find\_mean(){                 // функція середнього арифметичного масиву

    double suma {};

    double mean {};

    for (int i= 0; i < n; i++){

            suma += old[i];

        }

    mean = suma/n;

    return mean;

}

Код програми

2) Результат



Рисунок 2 результат

3) Посилання на файл програми у пул-запиті GitHub

Pull-[request](https://github.com/artificial-intelligence-department/ai_programming_playground/pull/683/files#diff-1b757905e8e8b4b811aaa3eb3fccf371ff15ea0b06d4cad98a59a7f00d4cab8a)

ssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssssss

Завдання №2 VNS Lab 5

1)Код програми

#include <iostream>

#include <vector>

using namespace std;

bool find\_column(const vector<vector<int>>&);

int totalSum {0};

int main() {

    int rows, columns;

    cout << "How much rows?: ";

    cin >> rows;

    cout << "How much columns: ";

    cin >> columns;

    vector<vector<int>> matrix(rows, vector<int>(columns));

    cout << "Enter matrix elements:\n";

    for (int i = 0; i < rows; ++i) {

        for (int j = 0; j < columns; ++j) {

            cin >> matrix[i][j];

        }

    }

    if (find\_column(matrix)) {

        cout << "Yes\n";

    } else {

        cout << "No\n";

    }

    return 0;

}

bool find\_column(const vector<vector<int>> &matrix) {

    for (const auto& row : matrix) {

        for (int elem : row) {

            totalSum += elem;

        }

    }

    for (int col = 0; col < matrix[0].size(); ++col) {

        int sumBefore = 0;

        for (int row = 0; row < matrix.size(); ++row) {

            sumBefore += matrix[row][col];

        }

        if (sumBefore > totalSum - sumBefore) {

            return true;

        }

    }

    return false;

}

Код програми

2) Результати

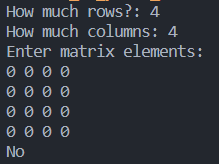
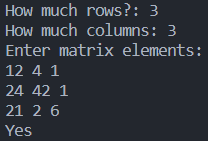


Рисунок 3 результат Рисунок 4 результат

3) Посилання на файл програми у пул-запиті GitH ub

Pull-[Request](https://github.com/artificial-intelligence-department/ai_programming_playground/pull/683/files#diff-48cb43f29fb15527bfe84d2266cd4d715db5605330b6f562b88246e80dadddb3)

Завдання №3 Algotester Lab 2

1)Код програми

#include <iostream>

#include <cmath>

using namespace std;

int main(){

    cout << "Enter array range: ";

    int N {};

    cin >> N;

    int r[N] {};

    cout << "Enter numbers for array: ";

    for (int i = 0; i < N; i++)

    {

        cin >> r[i];

    }

    int first{};

    int second{};

    int third{};

    int NewSize{0};

    int New[N] {};

     cout << "Enter numbers u want to remove from array: ";

     cin >> first >> second >> third;

    for (int i=0; i < N; i++){

        if ( r[i] != first && r[i] != second  &&  r[i] != third  ){

            New[NewSize]=r[i];

            NewSize++;

        }

     }

     if(NewSize==0)

        cout <<"0";

     else

     cout << NewSize-1 << endl;

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

    {

        cout << New[i]+New[i+1] <<" ";

    }

    return 0;

}

Код програми

2) Результат



Рисунок 5 результат в алготестері

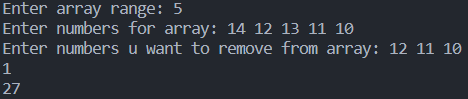


Рисунок 6 результат

3) Посилання на файл програми у пул-запиті GitHub

Pull-[Request](https://github.com/artificial-intelligence-department/ai_programming_playground/pull/683/files#diff-155ea64082fd7cabc9ee2e7cca673d7ca4615478f2c50a643bf519364be0f528)

Завдання 4. Algotester Lab 3

1)Код програми

#include <iostream>

using namespace std;

int same {0};

int unique {0};

int main(){

    cout << "Enter first array range: ";

    int N {};

    cin >> N;

    int first[N] {};

    cout << "Enter numbers for first array: ";

    for (int i = 0; i < N; i++)

    {

        cin >> first[i];

    }

cout << "Enter second array range: ";

    int M {};

    cin >> M;

    int second[M] {};

    cout << "Enter numbers for second array: ";

    for (int i = 0; i < M; i++)

    {

        cin >> second[i];

    }

    for (int i = 0; i < N; i++){

        for (int k = 0; k < M; k++){

            if (first[i] == second[k])

                same++;

      }

    }

    unique = (M+N)-same;

    cout << same << endl;

    cout << unique;

    return 0;

}

Код програми

1. Результати

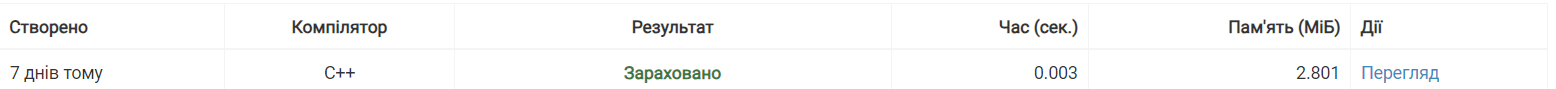


Рисунок 7 результат в алготестері

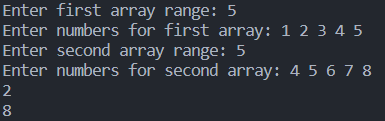


Рисунок 8 результат

3) Посилання на файл програми у пул-запиті GitHub

Pull-[Request](https://github.com/artificial-intelligence-department/ai_programming_playground/pull/683/files#diff-79949236f2ee51857d624fda3c2d8cd4c81c5367a1857faf9ef5e798faa9e2e2)

Завдання 5. Practice

1)Код програми

#include <iostream>

#include <vector>

#include <string>

using namespace std;

bool isPalindrome(int);

bool isPalindrome(const string&, int, int);

string word;

vector<int> numbers {};

int num {};

int normal {};

int reverse {};

int start {};

int finish {};

size\_t range {};

bool palindrom {false};

int main(){

    cout << "Enter word or an integer: ";

    cin >> word;

    start = 0;

    finish = word.size() - 1;

    cout << boolalpha << isPalindrome(word, start, finish) << endl;

return 0;

}

bool isPalindrome(int num){

    normal = num;

    while (num>0){

        int pos {};

        pos=num%10;

        num /= 10;

        numbers.push\_back(pos);

    }

    range = numbers.size();

    for (int i = 0; i < range; i++){

        reverse = reverse\*10 + numbers.at(i);

        }

    if (normal == reverse)

        palindrom = true;

    else if (normal != reverse)

        palindrom = false;

    return palindrom;

}

bool isPalindrome(const string &word, int start, int finish) {

    if (start >= finish) {

        return true;

    }

    if (word[start] != word[finish]) {

        return false;

    }

    return isPalindrome(word, start + 1, finish - 1);

}

Код програми

2) Результати

Рисунок 9 результат Рисунок 10 результат Рисунок 11 результат



Рисунок 12 результат

3) Посилання на файл програми у пул-запиті GitHub

Pull-[Request](https://github.com/artificial-intelligence-department/ai_programming_playground/pull/683/files#diff-ad1ec3d46928476e8263361dbd10c4b29d52bc123b4a2c4307f21d5c543819eb)

Завдання 6. Self-practice

1)Код програми

#include <iostream>

#include <vector>

#include <string>

using namespace std;

char letter\_convertor(char& );

void print(vector<int>);

void add(int);

void find\_mean(double suma = 0);

void smallest();

void largest();

char choice{};

vector <int> numbers;

int num {};

size\_t range {};

double mean {};

int main(){

    menu:

    {

            bool done = false;

        while(!done){

        range = numbers.size();

            cout << "P - Print numbers\n" <<

            "A - Add number\n" <<

            "M - Display mean of the numbers\n" <<

            "S - Display the smallest number\n" <<

            "L - Display the largest number\n" <<

            "Q - Quit\n" <<

            "Enter your choice: ";

            cin >> choice;

            letter\_convertor(choice);

        if( choice=='P' || choice=='A'|| choice=='M' ||

        choice=='S'|| choice=='L' || choice=='Q' )

                done = true;

        else

        cout <<"Oops... You probably wrote smth different. Please try again\n";

        }

}

switch (choice)

{

case 'P':

    print(numbers);

    goto menu;

case 'A':

    add(num);

    goto menu;

case 'M':

    find\_mean();

    goto menu;

case 'S':

    smallest();

    goto menu;

case 'L':

    largest();

    goto menu;

case 'Q':

    cout << "Have a nice day :)";

    break;

 }

}

char letter\_convertor(char &choice){

    if (choice >= 'a' && choice <= 'z'){

        choice = choice - 32;

    }

    return choice;

}

void print(vector<int> numbers){

    if (range == 0)

        cout << "there is no numbers to print" << endl;

    if (range != 0){

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

            cout << numbers.at(i) <<" ";

        }

    cout << endl;

    }

}

void add(int num){

    cout << "Enter number u want to add: ";

            cin >> num;

            numbers.push\_back(num);

            cout << num << " added" << endl;

}

void find\_mean(double suma){

        if(range==0){

            cout << "Unable to determine mean - list is empty" <<endl;

        }

        else {

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

            suma += numbers.at(i);

        }

    mean = suma/range;

    cout << mean << endl;

    }

}

void smallest(){

    if(range==0){

            cout << "Unable to determine the smallest number - list is empty" << endl;

            }

    else {

        int min = INT\_MAX;

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

        if (numbers.at(i)< min)

            min = numbers.at(i);

        }

    cout << min << endl;

    }

}

void largest(){

    if(range==0)

        cout << "Unable to the largest number - list is empty" << endl;

    else {

        int max = INT\_MIN;

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

            if (numbers.at(i)> max)

                max = numbers.at(i);

    }

        cout << max << endl;

    }

}

Код програми

2) Результати

P - Print numbers

A - Add number

M - Display mean of the numbers

S - Display the smallest number

L - Display the largest number

Q - Quit

Enter your choice: a

Enter number u want to add: 15

15 added

P - Print numbers

A - Add number

M - Display mean of the numbers

S - Display the smallest number

L - Display the largest number

Q - Quit

Enter your choice: a

Enter number u want to add: -1231

-1231 added

P - Print numbers

A - Add number

M - Display mean of the numbers

S - Display the smallest number

L - Display the largest number

Q - Quit

Enter your choice: a

Enter number u want to add: 192

192 added

P - Print numbers

A - Add number

M - Display mean of the numbers

S - Display the smallest number

L - Display the largest number

Q - Quit

Enter your choice: a

Enter number u want to add: 0

0 added

P - Print numbers

A - Add number

M - Display mean of the numbers

S - Display the smallest number

L - Display the largest number

Q - Quit

Enter your choice: p

15 -1231 192 0

P - Print numbers

A - Add number

M - Display mean of the numbers

S - Display the smallest number

L - Display the largest number

Q - Quit

Enter your choice: m

-256

P - Print numbers

A - Add number

M - Display mean of the numbers

S - Display the smallest number

L - Display the largest number

Q - Quit

Enter your choice: l

192

P - Print numbers

A - Add number

M - Display mean of the numbers

S - Display the smallest number

L - Display the largest number

Q - Quit

Enter your choice: s

-1231

P - Print numbers

A - Add number

M - Display mean of the numbers

S - Display the smallest number

L - Display the largest number

Q - Quit

Enter your choice: q

Good bye :)

Результат

3) Посилання на файл програми у пул-запиті GitHub

Pull-[Request](https://github.com/artificial-intelligence-department/ai_programming_playground/pull/683/files#diff-1a99c6e6f11c40e8712137eda8990848616c0230f5073f311e6bfd3f357f5406)

***Міт з командою***

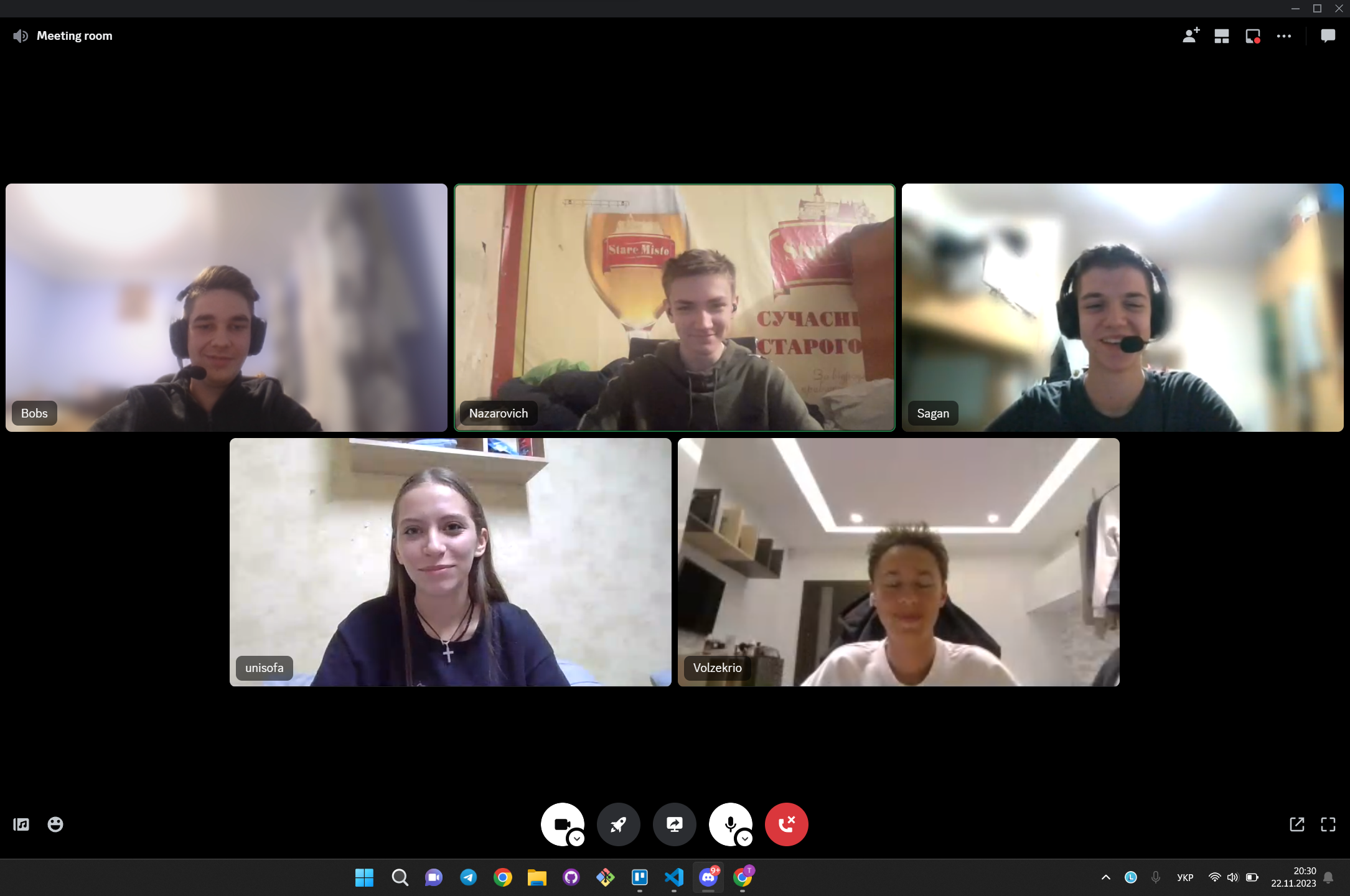


Рисунок 13 Міт з командою в Discord

# Висновок:

У процесі цього епіку я успішно оволодів основними конструкціями, такими одновимірні масиви, двовимірні масиви, алгоритми обробки. Виконуючи різноманітні завдання, такі як Algotester Lab 3 мені вдалося застосувати отримані знання на практиці. Це мій ще один крок у свiт програмування і впевнений, що ці набутi навички будуть корисними для подальшого навчання та програмування.