Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра систем штучного інтелекту



**Звіт**

про виконання

**Лабораторних та практичних робіт №2,3,7**

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

***з розділу***: «Цикли. Вкладені Цикли. Функції. Перевантаження функцій. Рекурсія.»

***Виконала:***

студент групи ШІ-13

Присліпська Софія Андріївна

# **Тема роботи:**

«Цикли. Вкладені Цикли. Функції. Перевантаження функцій. Рекурсія.»

# **Мета роботи:**

* Task 1 - Theory Education Activities
* Task 2 - Requirements management (understand tasks) and design activities (draw flow diagrams and estimate tasks 3-7)
* Task 3 - Lab# programming: VNS Lab 2
* Task 4 - Lab# programming: VNS Lab 3
* Task 5 - Lab# programming: VNS Lab 7
* Task 6 - Practice# programming: Class Practice Task
* Task 7 - Practice# programming: Self Practice Task
* Task 8 - Result Documentation Report and Outcomes Placement Activities (Docs and Programs on GitHub)
* Task 9 - Results Evaluation and Release

# **Теоретичні відомості:**

1. Теоретичні відомості з переліком важливих тем:

* Тема №1:Цикли та вкладені цикли
* Тема №2: Функції та їх перенавантаження
* Тема №3: Рекурсія

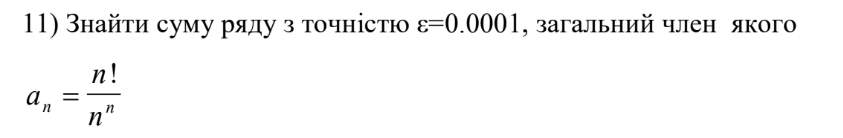
1. Індивідуальний план опрацювання теорії:

* Тема №1: Цикли та вкладені цикли.
  + Джерела Інформації
    - <https://www.bestprog.net/uk/2017/09/04/cycles-operators-of-the-cycle-for-while-do-while_ua/>
  + Що опрацьовано:
    - поняття циклів в програмуванні.
  + Статус: Ознайомлена
  + Початок опрацювання теми: 15.11
  + Звершення опрацювання теми:15.11
* Тема №2: Функції та їх перенавантаження
  + Джерела Інформації:
    - <https://acode.com.ua/urok-108-perevantazhennya-funktsij/>
    - <https://acode.com.ua/urok-15-funktsiyi-i-operator-return/>
    - <https://acode.com.ua/urok-16-parametry-i-argumenty-funktsij/>
    - <https://acode.com.ua/urok-104-peredacha-po-posylannyu/>
  + Що опрацьовано:
    - опрацювала функції та їх перенавантаження
  + Статус: Ознайомлена
  + Початок опрацювання теми:20.11
  + Звершення опрацювання теми: 20.11
* Тема №3:Рекурсія
  + Джерела Інформації:
    - <https://youtu.be/V7q9w_s0nns?feature=shared>
    - <https://www.bestprog.net/uk/2019/01/07/recursion-examples-of-tasks-solving-advantages-and-disadvantages-of-recursion-ua-2/>
  + Що опрацьовано:
    - опрацьовано матеріал про рекурсію.
  + Статус: Ознайомлена
  + Початок опрацювання теми:20.11
  + Звершення опрацювання теми: 20.11

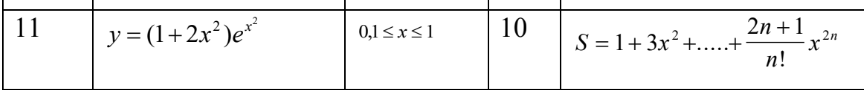
# **Виконання роботи:**

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

Завдання № 1 VNS Lab 2

* Варіант завдання:11
* 

Завдання № 2 VNS Lab 3

* Варіант завдання: 11
* Для х, що змінюється від a до b з кроком (b-a)/k, де (k=10), обчислити
* функцію f(x), використовуючи її розклад в степеневий ряд у двох випадках:
* а) для заданого n;
* б) для заданої точності ε (ε=0.0001).
* Для порівняння знайти точне значення функції.
* 

Завдання № 3 VNS Lab 7

* Варіант завдання:11
* 1)
* Написати функцію sum зі змінною кількістю параметрів, що знаходить суму
* заданих звичайних дробів. Написати викликаючу функцію main, що
* звертається до функції sum не менше трьох разів з кількістю параметрів 5, 10,
* 12.
* 2)Написати перевантажені функції й основну програму, що їх викликає:
* а) для множення десяткових дробів;
* б) для множення звичайних дробів.

Завдання № 4 Class Practice Task

# Менеджмент бібліотеки

# Задача

Ви створюєте просту програму керування бібліотекою. Книги в бібліотеці є, користувачі можуть їх взяти або повернути.

### Програма повинна вміти

* Перерахувати всі книги.
* Дозволити взяти книгу (за наявності).
* Дозволити повернення книги.

### Структури даних

* Використовуйте масив або вектор для зберігання назв книг.
* Використовуйте інший масив або вектор для збереження стану доступності кожної книги.

## Мета Задачі

Навчитися користуватися операторами циклів та функцією переходу на мітку:

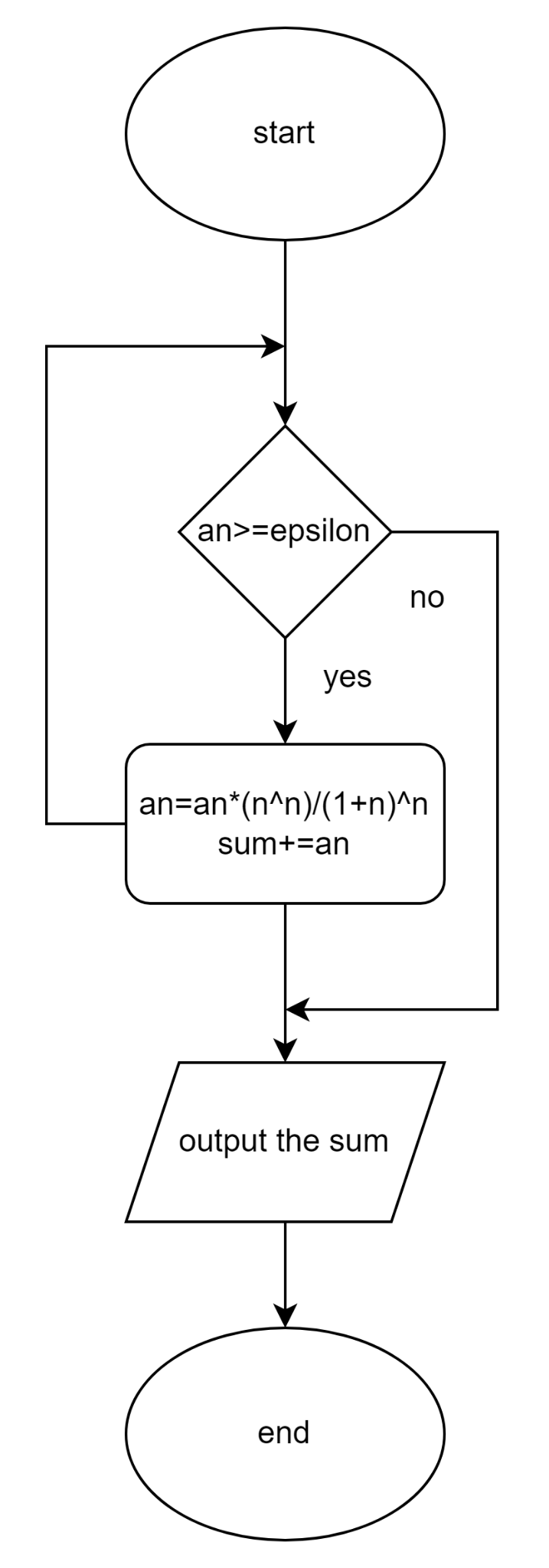
1. for( ) { … }
2. for each
3. while( ) { … }
4. do { … } while( )
5. go to

### Вимоги:

1. while: продовжувати працювати, доки користувач не вирішить вийти.
2. do while: Після кожної операції (позичити, повернути, перерахувати) запитуйте користувача, чи хоче він виконати іншу операцію. Якщо так, поверніться назад.
3. for: список усіх книг за допомогою циклу.
4. for each: перевірити наявність кожної книги.
5. goto: якщо користувач вводить неправильний вибір, використовуйте goto, щоб перенаправити його до головного меню.

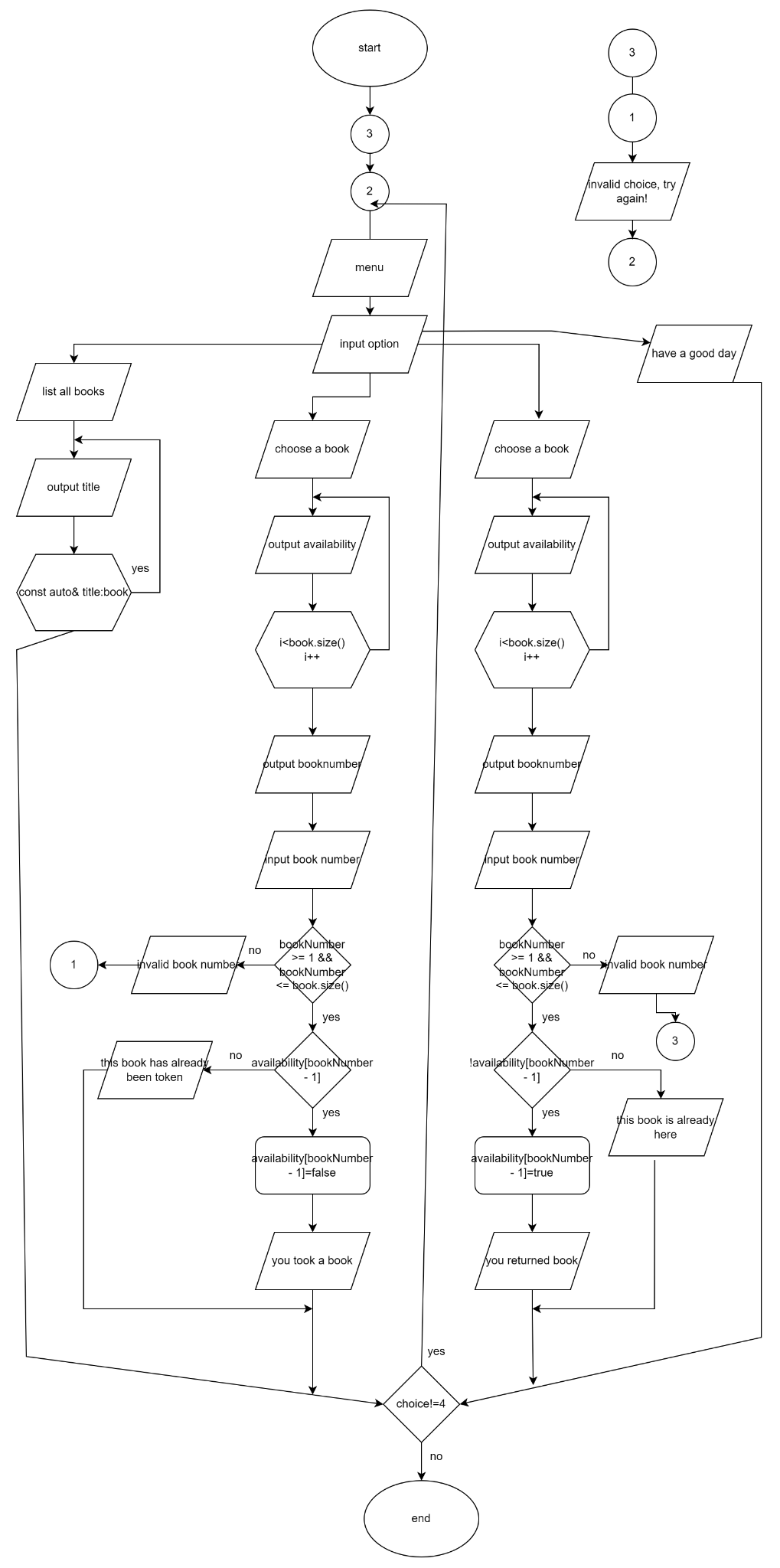
## **2. Дизайн та планована оцінка часу виконання завдань:**

Програма № 1 VNS Lab 2

* Блок-схема
* 

Завдання № 4 Class Practice Task

* Блок-схема



* Планований час на реалізацію

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

Завдання №1 VNS Lab 2

#include <iostream>

#include <cmath>

using namespace std;

int main()

{

int n=1;

double sum=0;

double epsilon = 0.0001;

double an = 1; // n+1 член послідовності

while (an >= epsilon)

{

sum += an;

n++;

an = an \* (pow(n, n)) / (pow(1 + n, n));

}

cout << sum << endl;

return 0;

}

Підпис та № 1до блоку з кодом програми

Завдання №2 VNS Lab 3

#include <iostream>

#include <cmath>

using namespace std;

double function(double x)

{

return (1 + 2 \* pow(x, 2)) \* exp(pow(x, 2));

}

int main()

{

double a = 0.1;

double b = 1.0;

int k = 10; // Кількість кроків

int n = 10;

double epsilon = 0.0001;

double step = (b - a) / k; // Розрахунок кроку

// Цикл для зміни параметра x

for (double x = a; x <= b; x += step)

{

double sn = 1; // Значення суми для заданого n

double se = 1; // Значення суми для заданої точності

double an = 1; // Початковий член ряду для заданого n

double ae = 1; // Початковий член ряду для заданої точності

int i = 1;

// Цикл для обчислення суми за заданим n

while (i <= n)

{

an \*= (2 \* i + 1) \* pow(x, 2) / (i \* (i + 1)); // Обчислення кожного члена ряду

sn += an;

i++;

}

i = 1;

// Цикл для обчислення суми за заданою точністю epsilon

while (ae >= epsilon)

{

ae \*= (2 \* i + 1) \* pow(x, 2) / (i \* (i + 1)); // Обчислення кожного члена ряду

se += ae;

i++;

}

// Виведення результатів

cout << "X:\t " << x << " SN:\t " << sn << " SE:\t " << se << " Y:\t " << function(x) << "\n";

}

return 0;

}

Підпис та № 2 до блоку з кодом програми

Завдання №3.1 VNS Lab 7

#include <iostream>

#include <cstdarg>

using namespace std;

// Функція для обчислення суми звичайних дробів зі змінною кількістю параметрів

double sum(int count, ...) {

va\_list args;

double sum = 0;

va\_start(args, count);

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

sum += va\_arg(args, double);

}

va\_end(args);

return sum;

}

int main() {

// Виклики функції sum зі змінною кількістю параметрів

double result1 = sum(5, 1.5, 2.0, 3.5, 4.0, 5.5);

double result2 = sum(10, 1.1, 2.2, 3.3, 4.4, 5.5, 6.6, 7.7, 8.8, 9.9, 10.1);

double result3 = sum(12, 1.2, 2.3, 3.4, 4.5, 5.6, 6.7, 7.8, 8.9, 9.0, 10.1, 11.2, 12.3);

cout << "Result 1: " << result1 << endl;

cout << "Result 2: " << result2 << endl;

cout << "Result 3: " << result3 << endl;

return 0;

}

Підпис та № 3.1 до блоку з кодом програми

Завдання №3.2 VNS Lab 7

#include <iostream>

using namespace std;

// Перевантажена функція для множення десяткових дробів

float multiply(float a, float b)

{

return a \* b;

}

// Перевантажена функція для множення звичайних дробів

float multiply(int num1, int den1, int num2, int den2)

{

return (num1 \* num2) / (den1 \* den2);

}

int main()

{

// множення десяткових дробів

float result1 = multiply(3.5, 2.0);

cout << "Result of multiplying decimals: " << result1 << endl;

// множення звичайних дробів

float result2 = multiply(5, 10, 20, 10);

cout << "Result of multiplying fractions: " << result2 << endl;

return 0;

}

Підпис та № 3.2 до блоку з кодом програми

Завдання №4 Class Practice Task

#include <iostream>

#include <vector>

#include <string>

using namespace std;

int main() {

vector<string> book = {"The Night in Lisbon", "Misery", "The Great Gatsby ", "Little Women"}; // Список назв книг

vector<bool> availability = {true, true, true, true}; // Список доступності книг

char choice;

do {

cout << "Menu:" << endl

<< "1. List all books" << endl

<< "2. Take the book" << endl

<< "3. Return the book" << endl

<< "4. Exit" << endl

<< "Select an option: "<<endl;

cin >> choice;

switch (choice) {

case '1': // Перерахувати всі книги

cout << "list all books" << endl;

for (const auto& title : book) {

cout << title << endl;

}

break;

case '2': { // Взяти книгу

cout << "Choose a book by number: " << endl;

for (int i = 0; i < book.size(); ++i) {

cout << i + 1 << ". " << book[i] << " - ";

cout << (availability[i] ? "available" : "unavailable") <<endl;

}

int bookNumber;

cout << "book number: ";

cin >> bookNumber;

if (bookNumber >= 1 && bookNumber <= book.size()) {

if (availability[bookNumber - 1]) {

availability[bookNumber - 1] = false;

cout << "you took a book '" << book[bookNumber - 1] << "'" << endl;

} else {

cout << "this book has already been token" << endl;

}

} else {

cout << "invalid book number" << endl;

goto invalidChoice;

}

break;

}

case '3': { // Повернути книгу

cout << "Choose a book by number: " << endl;

for (int i = 0; i < book.size(); ++i) {

cout << i + 1 << ". " << book[i] << " - ";

cout << (availability[i] ? "available" : "unavailable") << endl;

}

int bookNumber;

cout << "book number: ";

cin >> bookNumber;

if (bookNumber >= 1 && bookNumber <= book.size()) {

if (!availability[bookNumber - 1]) {

availability[bookNumber - 1] = true;

cout << "you returned a book '" << book[bookNumber - 1] << "'" << endl;

} else {

cout << "This book is already here" << endl;

}

} else {

cout << "invalid book number" << endl;

goto invalidChoice;

}

break;

}

case '4': // Вийти

cout << "have a good day!" << endl;

break;

default:

invalidChoice:

cout << "invalid choice. Try again!" << endl;

break;

}

} while (choice != '4');

return 0;

}

Підпис та № 4 до блоку з кодом програми

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

Завдання №1 Деталі по виконанню і тестуванню програми

1.75964

Підпис та №1 до блоку з виконанням та тестуванням програми

Час затрачений на виконання завдання:3 години

Завдання №2 Деталі по виконанню і тестуванню програми

X: 0.10000 SN: 1.01513 SE: 1.01513 Y: 1.03025

X: 0.19000 SN: 1.05581 SE: 1.05581 Y: 1.11161

X: 0.28000 SN: 1.12565 SE: 1.12565 Y: 1.25114

X: 0.37000 SN: 1.23077 SE: 1.23077 Y: 1.46068

X: 0.46000 SN: 1.38099 SE: 1.38099 Y: 1.75858

X: 0.55000 SN: 1.59140 SE: 1.59140 Y: 2.17195

X: 0.64000 SN: 1.88504 SE: 1.88504 Y: 2.74011

X: 0.73000 SN: 2.29730 SE: 2.29730 Y: 3.51985

X: 0.82000 SN: 2.88321 SE: 2.88319 Y: 4.59331

X: 0.91000 SN: 3.72986 SE: 3.72985 Y: 6.07995

X: 1.00000 SN: 4.97777 SE: 4.97777 Y: 8.15485

Підпис та №2до блоку з виконанням та тестуванням програми

Час затрачений на виконання завдання

Завдання №3.1 тестування програми

Result 1: 16.5

Result 2: 59.6

Result 3: 83

Підпис та №3.1 до блоку з виконанням та тестуванням програми

Час затрачений на виконання завдання:3 години

Завдання №3.2 тестування програми

Result of multiplying decimals: 7

Result of multiplying fractions: 1

Підпис та №3.2 до блоку з виконанням та тестуванням програми

Час затрачений на виконання завдання:2 години

Завдання №4

Menu:

1. List all books

2. Take the book

3. Return the book

4. Exit

Select an option:

2

Choose a book by number:

1. The Night in Lisbon - available

2. Misery - available

3. The Great Gatsby - available

4. Little Women - available

book number: 4

you took a book 'Little Women'

Menu:

1. List all books

2. Take the book

3. Return the book

4. Exit

Select an option:

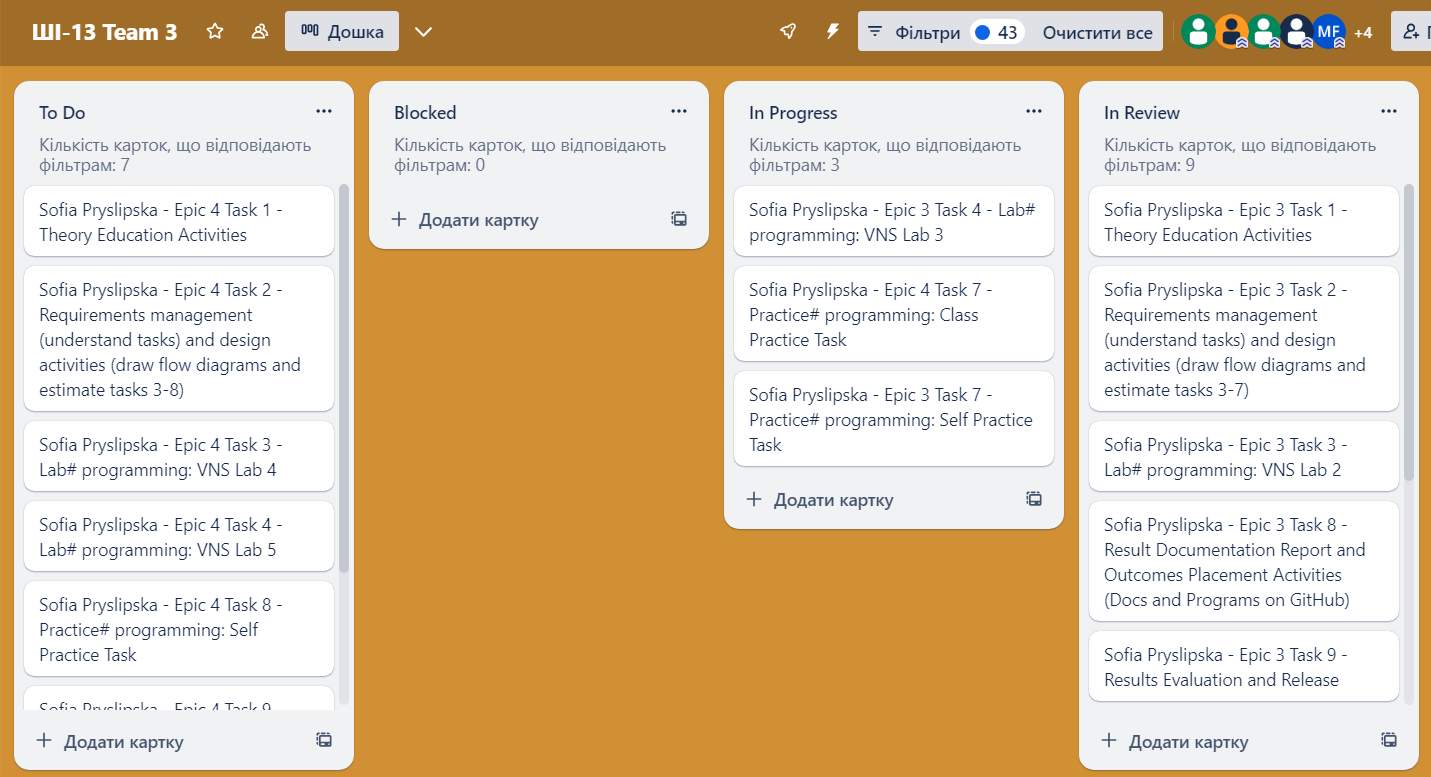
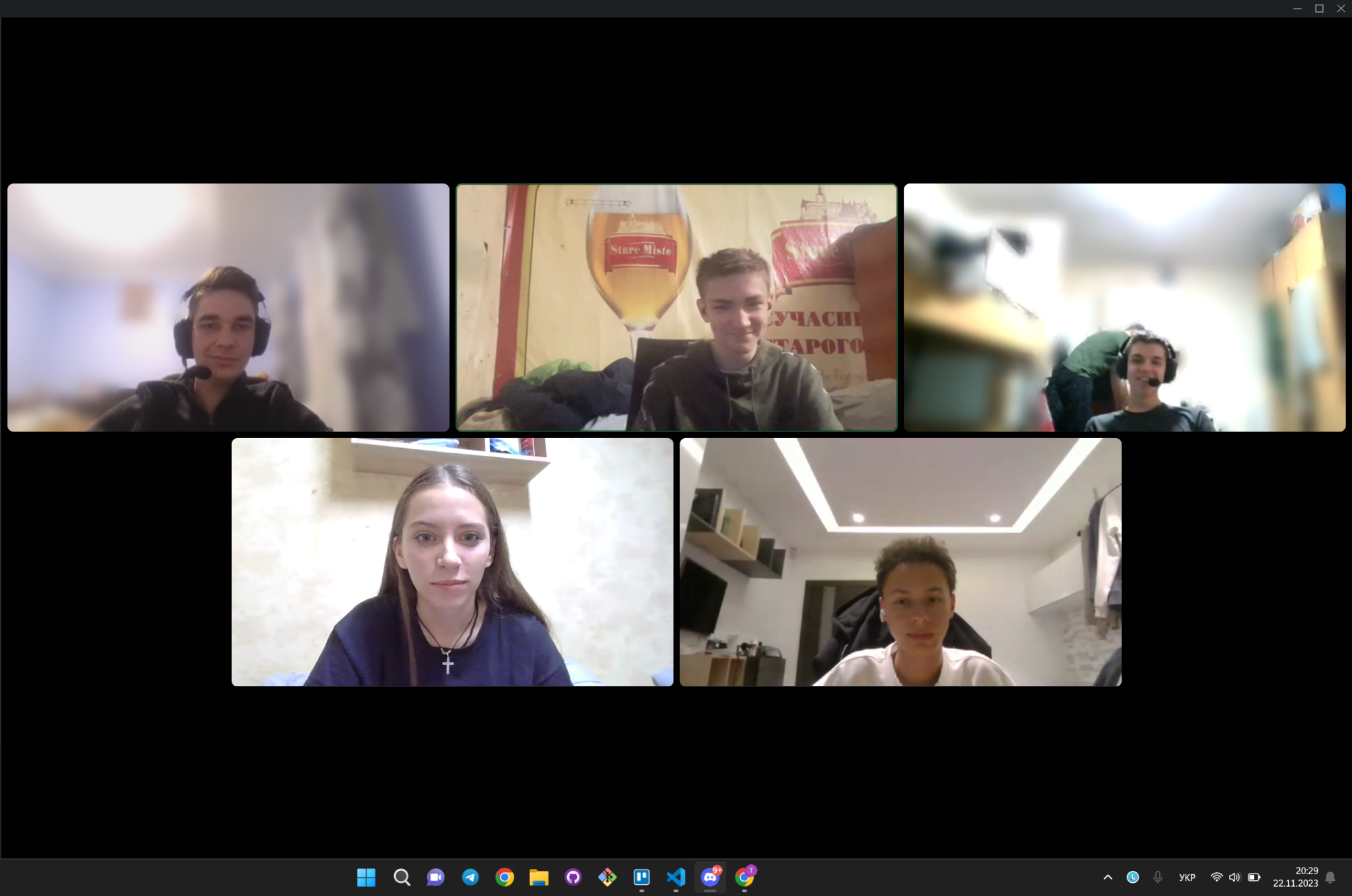
4

have a good day!

Підпис та №4 до блоку з виконанням та тестуванням програми

Час затрачений на виконання завдання: тиждень

## **5. Кооперація з командою:**

* 
* 

# **Висновки:**

при виконанні 2,3,7 лабораторних робіт я засвоїла багато нового матеріалу по таких темах Цикли, Вкладені Цикли, Функції, Перевантаження функцій, Рекурсія. Написала програму, яка виконує функції звичайної бібліотеки, також програму яка обчислює добуток звичайних та десяткових дробів. Ще одне завдання було обчислити задану функцію. Також потрібно було обчислити суму ряду використовуючи рекурентну формулу.