Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра систем штучного інтелекту



**Звіт**

про виконання

**Лабораторних та практичних робіт № 3**

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

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

***Виконав:***

студент групи ШІ-14

Мруць Остап Мар’янович

# **Тема роботи:**

Цикли, використання циклів у С++. Вкладені цикли. Обчислення суми рядів за допомогою циклів. Функції та їх перевантаження. Рекурсія функції. Створення функції для обчислення виразу та функції зі змінною кількістю параметрів.

# **Мета роботи:**

Ознайомитися, використати та опрацювати цикли for, while, do while у С++. Ознайомитися та опрацювати функції та їхню організацію, перевантаження функції та рекурсію.

# **Теоретичні відомості:**

1. Теоретичні відомості з переліком важливих тем:

* Тема №1: Цикли. Вкладені цикли
* Тема №2: Оператори continue, break, goto.
* Тема №3: Функція.
* Тема №4: Перевантаження функції.
* Тема №5: Рекурсія

1. Індивідуальний план опрацювання теорії:

* Тема №1:Цикли. Вкладені цикли .
  + Джерела Інформації
    - Відео.

<https://www.youtube.com/watch?v=ckJtOMcIxyU&ab_channel=%D0%91%D0%BB%D0%BE%D0%B3%D0%B0%D0%BD%E2%A6%81%D0%A3%D1%80%D0%BE%D0%BA%D0%B8%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F>

<https://www.youtube.com/watch?v=QXaSSIjVor8&ab_channel=%D0%91%D0%BB%D0%BE%D0%B3%D0%B0%D0%BD%E2%A6%81%D0%A3%D1%80%D0%BE%D0%BA%D0%B8%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F>

<https://www.youtube.com/watch?v=pBhaBdXWMmU&t=575s&ab_channel=%D0%91%D0%BB%D0%BE%D0%B3%D0%B0%D0%BD%E2%A6%81%D0%A3%D1%80%D0%BE%D0%BA%D0%B8%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F>

* + - Стаття.

<http://cpp.dp.ua/operatory-tsyklu/>

<https://acode.com.ua/urok-72-tsykl-for/>

<https://acode.com.ua/urok-70-tsykl-while/>

* + Що опрацьовано:
    - Цикли у С++: for, while, do while. Різницю між цими циклами. Вкладені цикли
  + Статус: Ознайомлений
* Тема №2: Оператори continue, break, goto.
  + Джерела Інформації:
    - Відео.

<https://www.youtube.com/watch?v=rj1OLsBKazA&t=399s&ab_channel=%D0%91%D0%BB%D0%BE%D0%B3%D0%B0%D0%BD%E2%A6%81%D0%A3%D1%80%D0%BE%D0%BA%D0%B8%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F>

<https://www.youtube.com/watch?v=UY295pIdeoQ&t=337s&ab_channel=%D0%91%D0%BB%D0%BE%D0%B3%D0%B0%D0%BD%E2%A6%81%D0%A3%D1%80%D0%BE%D0%BA%D0%B8%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F>

* + - Стаття.

<https://acode.com.ua/urok-73-operatory-break-i-continue/#toc-1>

<https://acode.com.ua/urok-69-operator-goto/>

* + Що опрацьовано:
    - Оператори переходу break, goto, continue та як вони застосовуються в циклах
  + Статус: Ознайомлений
* Тема №3:Функція.
  + Джерела Інформації:
    - Відео.

<https://www.youtube.com/watch?v=G8P6SvdqU9s&t=1663s&ab_channel=%D0%91%D0%BB%D0%BE%D0%B3%D0%B0%D0%BD%E2%A6%81%D0%A3%D1%80%D0%BE%D0%BA%D0%B8%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F>

* + - Стаття.

<https://acode.com.ua/urok-15-funktsiyi-i-operator-return/>

* + Що опрацьовано:
    - Функції С++, типи повернення функцій, прототипи і визначення функції
  + Статус: Ознайомлений
* Тема №4 Перевантаження функції.
  + Джерела Інформації:
    - Відео.

<https://www.youtube.com/watch?v=3KJfisev6SI&ab_channel=%D0%91%D0%BB%D0%BE%D0%B3%D0%B0%D0%BD%E2%A6%81%D0%A3%D1%80%D0%BE%D0%BA%D0%B8%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F>

* + - Стаття.

<https://acode.com.ua/urok-108-perevantazhennya-funktsij/>

* + Що опрацьовано:
    - Перевантаження функції
  + Статус: Ознайомлений
* Тема №5 Рекурсія.
  + Джерела Інформації:
    - Відео.

<https://www.youtube.com/watch?v=V7q9w_s0nns&ab_channel=%D0%91%D0%BB%D0%BE%D0%B3%D0%B0%D0%BD%E2%A6%81%D0%A3%D1%80%D0%BE%D0%BA%D0%B8%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F>

* + - Стаття.

<https://acode.com.ua/urok-113-rekursiya-i-chysla-fibonachchi/>

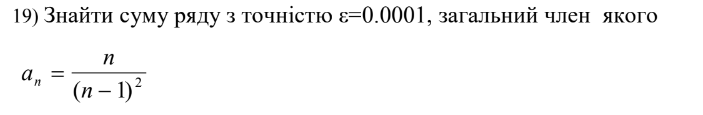
<http://cpp.dp.ua/rekursyvni-funktsiyi/>

* + Що опрацьовано:
    - Що таке рекурсія. Рекурсивні функції
  + Статус: Ознайомлений

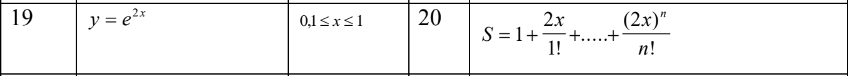
# **Виконання роботи:**

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

Завдання №1 VNS Lab 2 - Task 1

* Варіант 19
* 

Завдання №2 VNS Lab 3 - Task 1

* Варіант 19
* 

Завдання №3 VNS Lab 7

* Варіант завдання 19
* Деталі завдання

1. Написати функцію belong зі

змінною кількістю параметрів, що визначає скільки точок з координатами

(х,у) належать заданому кругу.

Написати викликаючу функцію main, що

звертається до функції belong не менше трьох разів з кількістю параметрів 3,

9, 11.

2. Написати перевантажені функції для додавання дійсних чисел і додавання комплексних чисел й основну програму, що їх викликає.

а) для масиву цілих чисел знаходить мінімальний елемент;

б) для рядка знаходить довжину найкоротшого слова .

* Важливі деталі для врахування в імплементації програми

Потрібно використати перевантаження функції.

Завдання №5 Class Practice Work

* Без варіантів
* Проста програма керування бібліотекою. Книги в бібліотеці є, користувачі можуть їх взяти або повернути.

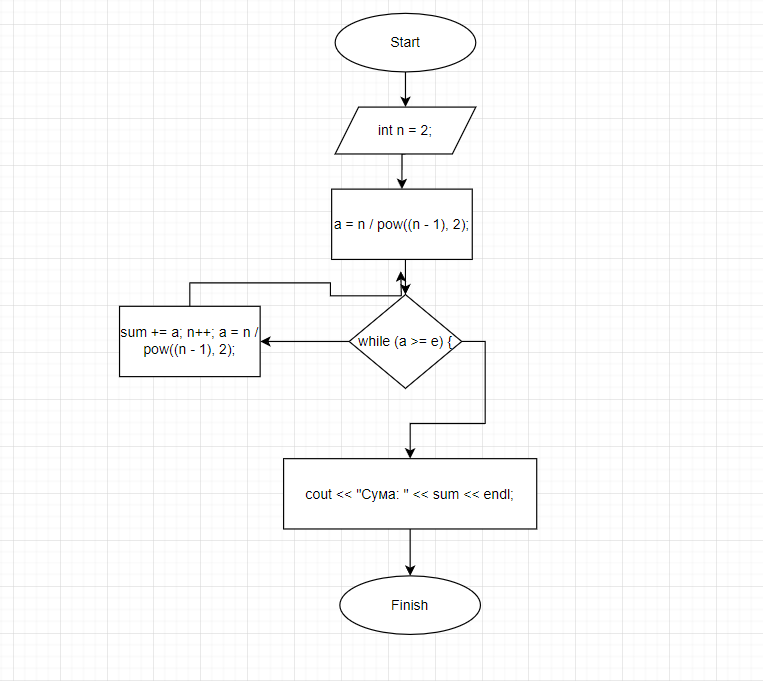
Завдання №6 Self Practice Work

У книжковому магазині є n книг різних цін. Петро має k гривень. Напиши програму, яка визначить максимальну кількість книг, які може собі дозволити Петро на свої гроші.

## **2. Дизайн завдань:**

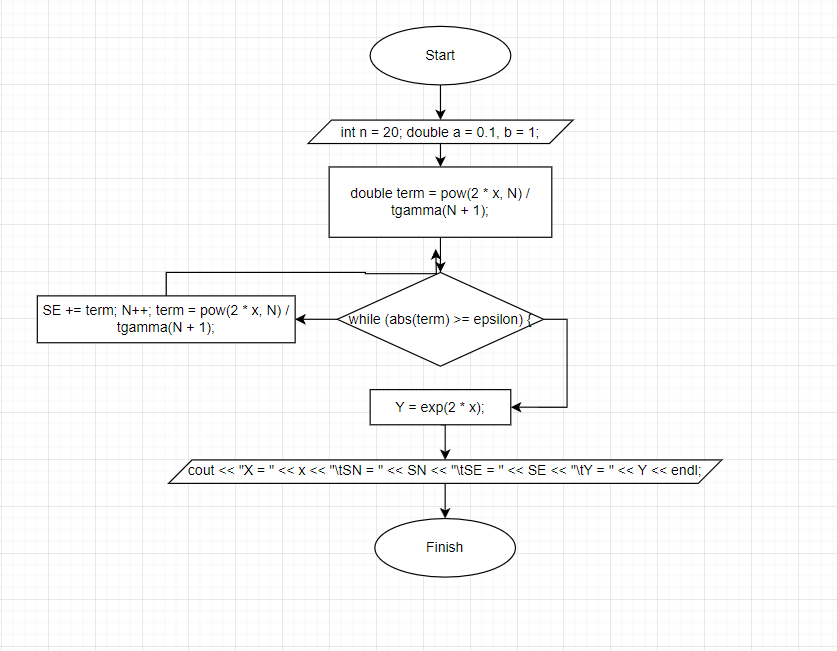
Програма №1 VNS Lab 2 - Task 1

* Блок-схема



*Image 1**. Блок-схема до програми №1 VNS Lab 2 Task 1*

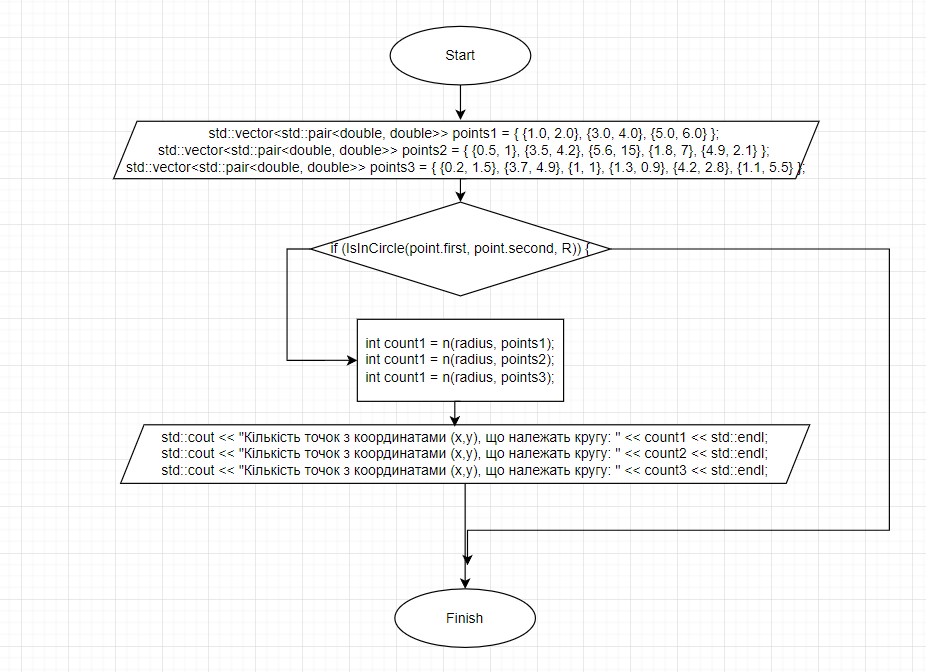
Програма №2 VNS Lab 3 - Task 1

* Блок-схема
* 

*Image 2. Блок-схема до програми №2 VNS Lab 3 Task 1*

Програма №3 VNS Lab 7 - Task 1

* Блок-схема

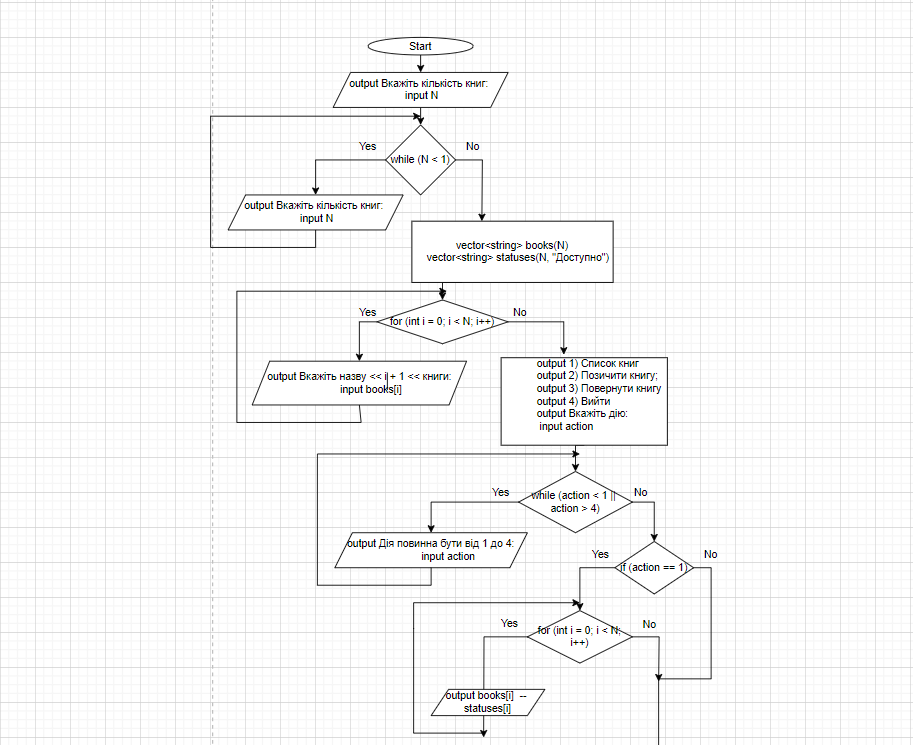


*Image 3. Блок-схема до програми №3 VNS Lab 7 Task 1*

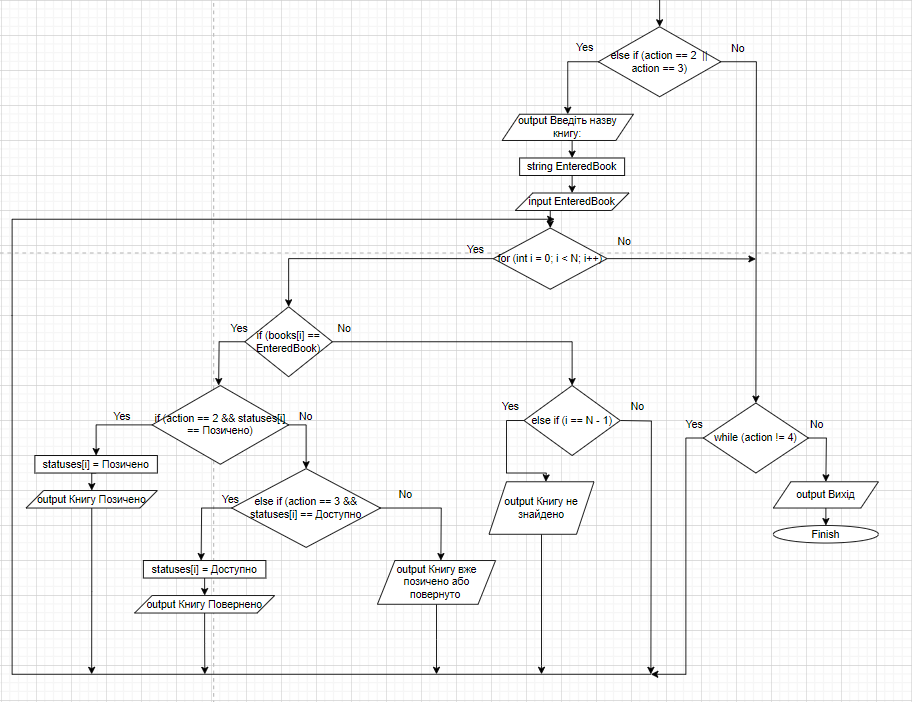
* Важливі деталі для врахування в імплементації. Організувано перевантажені функції

Програма №5 Class Practice Work

* Блок-схема



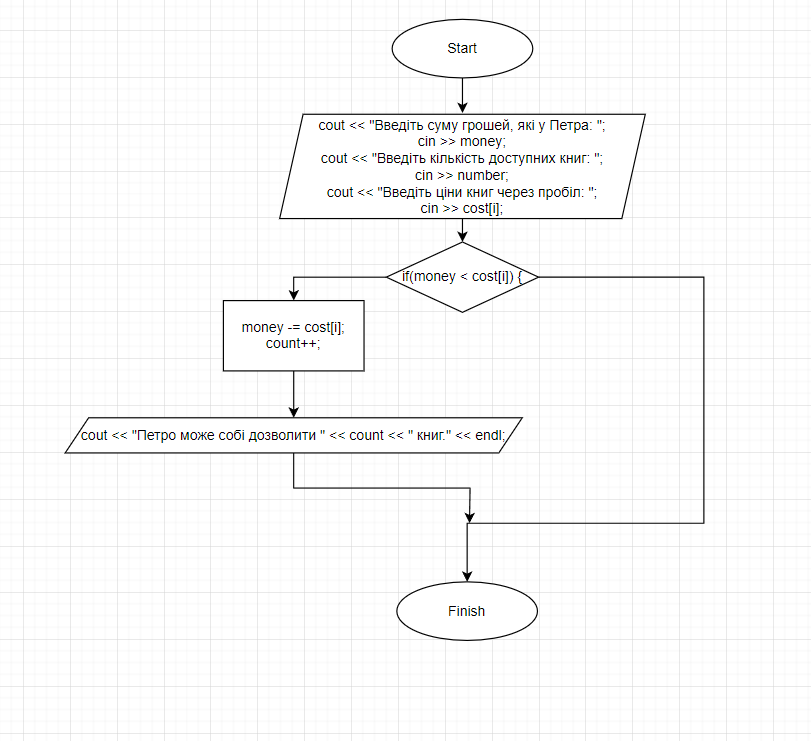
*Image 10. Блок-схема до програми №5 Class Practice Work*



*Image 11. Блок-схема до програми №5 Class Practice Work*

Програма №6 Self Practice Work

* Блок-схема



*Image 15. Блок-схема до програми №6 Self Practice Wor**k*

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

Використано налаштування з попередніх лабораторних робіт

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

Завдання №1 VNS Lab 2 - Task 1

Використовуючи оператор циклу, знайти суму ряду з точністю 0,0001

#include <iostream>

#include <cmath>

using namespace std;

int main() {

int n = 2;

double a, sum = 1, e = 0.0001;

a = n / pow((n - 1), 2);

while (a >= e) {

sum += a;

n++;

a = n / pow((n - 1), 2);

}

cout << "Сума: " << sum << endl;

return 0;

}

*Блок №2. Код до програми №1 VNS Lab 2 - Task 1*

Завдання №2 VNS Lab 3 - Task 1

Обчислити функцію f(x), використовуючи її розклад в степеневий ряд у двох випадках:

а) для заданого n = 20;

б) для заданої точності ε (ε=0.0001).

#include <iostream>

#include <cmath>

using namespace std;

int main() {

int n = 20;

double SN, SE, Y, epsilon = 0.0001;

double a = 0.1, b = 1;

for (double x = a; x <= b; x += 0.1) {

SN = 0;

SE = 0;

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

double term = pow(2 \* x, i) / tgamma(i + 1);

SN += term;

}

int N = 0;

double term = pow(2 \* x, N) / tgamma(N + 1);

while (abs(term) >= epsilon) {

SE += term;

N++;

term = pow(2 \* x, N) / tgamma(N + 1);

}

Y = exp(2 \* x);

cout << "X = " << x << "\tSN = " << SN << "\tSE = " << SE << "\tY = " << Y << endl;

}

return 0;

}

*Блок №3. Код до програми №2 VNS Lab 3 - Task 1*

Завдання №3 VNS Lab 7 - Task 1

Написати функцію (або макровизначення), що визначає чи належить точка з

координатами (х,у) кругу з заданим радіусом R.

#include <iostream>

#include <cmath>

#include <vector>

bool IsInCircle(double x, double y, double R) {

double distance = sqrt(x \* x + y \* y);

return distance <= R;

}

int n(double R, const std::vector<std::pair<double, double>>& points) {

int count = 0;

for (const auto& point : points) {

if (IsInCircle(point.first, point.second, R)) {

count++;

}

}

return count;

}

int main() {

double radius = 5.0;

std::vector<std::pair<double, double>> points1 = { {1.0, 2.0}, {3.0, 4.0}, {5.0, 6.0} };

int count1 = n(radius, points1);

std::cout << "Кількість точок з координатами (х,у), що належать кругу: " << count1 << std::endl;

std::vector<std::pair<double, double>> points2 = { {0.5, 1}, {3.5, 4.2}, {5.6, 15}, {1.8, 7}, {4.9, 2.1} };

int count2 = n(radius, points2);

std::cout << "Кількість точок з координатами (х,у), що належать кругу: " << count2 << std::endl;

std::vector<std::pair<double, double>> points3 = { {0.2, 1.5}, {3.7, 4.9}, {1, 1}, {1.3, 0.9}, {4.2, 2.8}, {1.1, 5.5} };

int count3 = n(radius, points3);

std::cout << "Кількість точок з координатами (х,у), що належать кругу: " << count3 << std::endl;

return 0;

}

*Блок №4. Код до програми №3 VNS Lab 7 - Task 1*

Завдання №4 VNS Lab 7 - Task 2

Написати перевантажені функції й основну програму, що їх викликає.

а) для масиву цілих чисел знаходить мінімальний елемент;

б) для рядка знаходить довжину найкоротшого слова.

##include <iostream>

#include <string>

#include <vector>

int Min(const std::vector<int>& arr) {

int min = arr[0];

for (size\_t i = 1; i < arr.size(); ++i) {

if (arr[i] < min) {

min = arr[i];

}

}

return min;

}

size\_t Min(const std::string& str) {

size\_t shortest = str.length();

size\_t currentLength = 0;

for (char c : str) {

if (c != ' ') {

++currentLength;

} else {

shortest = std::min(shortest, currentLength);

currentLength = 0;

}

}

shortest = std::min(shortest, currentLength);

return shortest;

}

int main() {

std::vector<int> numbers = {5, 3, 9, 1, 7};

std::cout << "Мінімальний елемент у масиві: " << Min(numbers) << std::endl;

std::string text = "The shortest word is";

std::cout << "Довжина найкоротшого слова: " << Min(text) << std::endl;

return 0;

}

*Блок №5. Код до програми №4 VNS Lab 7 - Task 2*

Завдання №5 Class Practice Work.

Проста програма керування бібліотекою. Книги в бібліотеці є, користувачі можуть їх взяти або повернути.

#include <iostream>

#include <string>

#include <vector>

using namespace std;

void displayBooks(const vector<string> &books, const vector<string> &statuses) {

cout << "List of books:\n";

for (size\_t i = 0; i < books.size(); ++i) {

cout << books[i] << " -- " << statuses[i] << "\n";

}

}

int main() {

int numBooks, action;

cout << "Enter the number of books: ";

cin >> numBooks;

while (numBooks < 1) {

cout << "Enter a valid number of books (1 or more): ";

cin >> numBooks;

}

vector<string> books(numBooks);

vector<string> statuses(numBooks, "Available");

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

cout << "Enter book " << i + 1 << ": ";

cin >> books[i];

}

do {

cout << "\n1) List of books\n";

cout << "2) Take a book\n";

cout << "3) Return a book\n";

cout << "4) Exit\n\n";

cout << "Choose your action: ";

cin >> action;

switch (action) {

case 1:

displayBooks(books, statuses);

break;

case 2:

case 3: {

cout << "Enter a book: ";

string inputBook;

cin >> inputBook;

bool bookFound = false;

for (size\_t i = 0; i < books.size(); ++i) {

if (books[i] == inputBook) {

bookFound = true;

if (action == 2 && statuses[i] == "Available") {

statuses[i] = "Borrowed";

cout << "Book " << books[i] << " borrowed successfully.\n";

} else if (action == 3 && statuses[i] == "Borrowed") {

statuses[i] = "Available";

cout << "Book " << books[i] << " returned successfully.\n";

} else {

cout << "Book not available for the chosen action.\n";

}

break;

}

}

if (!bookFound) {

cout << "Book not found.\n";

}

break;

}

case 4:

cout << "Exiting the program.\n";

break;

default:

cout << "Choose a valid action (1 to 4).\n";

break;

}

} while (action != 4);

return 0;

}

*Блок №6. Код до програми №5 Class Practice Work*

Завдання №6 Self Practice Work

У книжковому магазині є n книг різних цін. Петро має k гривень. Напиши програму, яка визначить максимальну кількість книг, які може собі дозволити Петро на свої гроші.

#include <iostream>

using namespace std;

int main() {

int cost[100], number, money;

cout << "Введіть суму грошей, які у Петра: ";

cin >> money;

cout << "Введіть кількість доступних книг: ";

cin >> number;

cout << "Введіть ціни книг через пробіл: ";

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

cin >> cost[i];

}

int count = 0;

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

if(money < cost[i]) {

break;

} else {

money -= cost[i];

count++;

}

}

cout << "Петро може собі дозволити " << count << " книг." << endl;

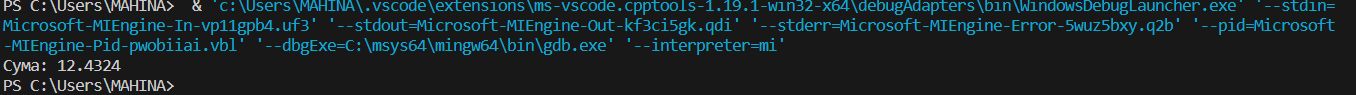
return 0;

}

*Блок №8. Код до програми №7 Self Practice Work*

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

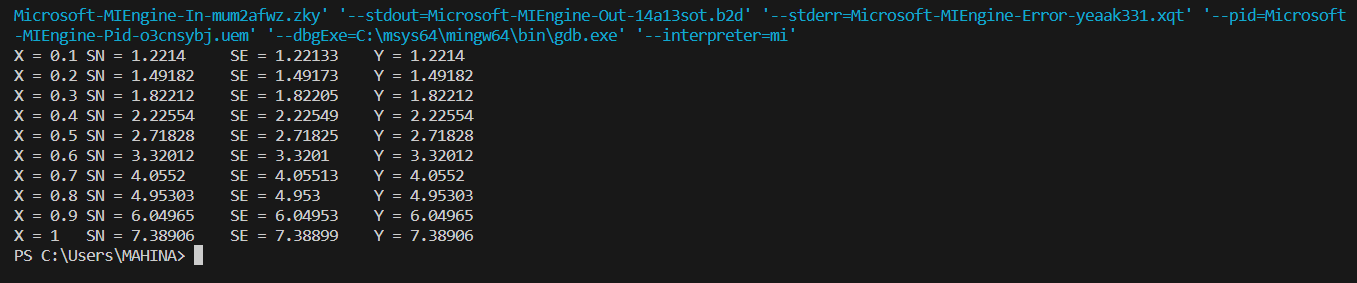
Завдання №1 VNS Lab 2 - Task 1



*Image 18**. Результати виконання завдання №1 VNS Lab 2 - Task 1*

Завдання №2 VNS Lab 3 - Task 1

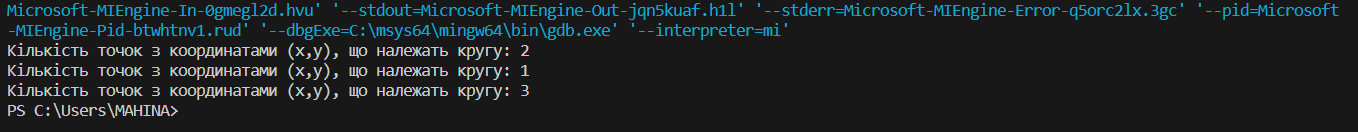
Програма обчислила суми показникового ряду (при n=20, де n - к-ть елементів, і зі заданою точністю 0,0001) і точне значення функції при певному X.



*Image 19. Результати виконання завдання №2 VNS Lab 3 - Task 1*

Завдання №3 VNS Lab 7 - Task 1

При викликах функції з різною кількістю параметрів, програма перетворює числа з двійкової системи в трійкову.



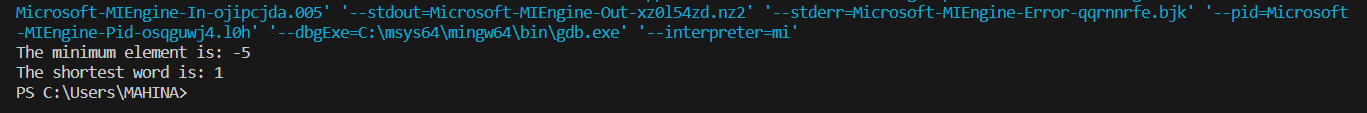
*Image 19. Результати виконання завдання №3 VNS Lab 7 - Task 1*

Завдання №4 VNS Lab 7 - Task 2

Програма рахує перевантажені функції

а) для масиву цілих чисел знаходить мінімальний елемент;

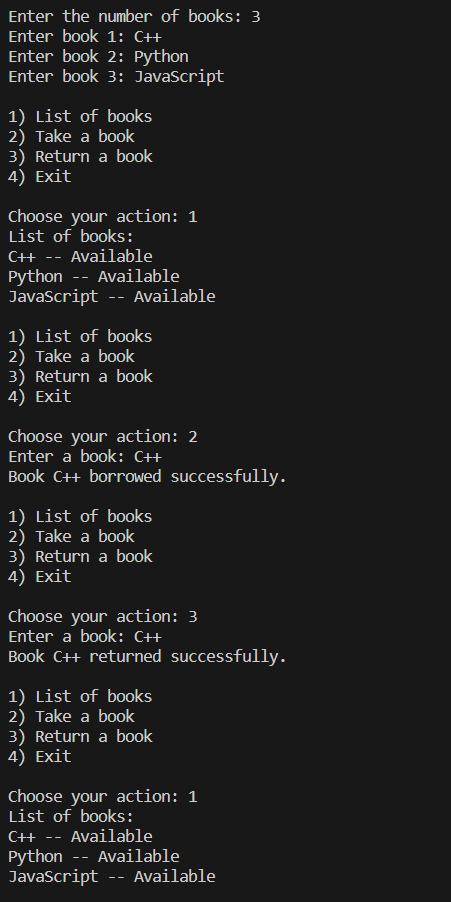
б) для рядка знаходить довжину найкоротшого слова .



*Image 20. Результати виконання завдання №4 VNS Lab 7 - Task 2*

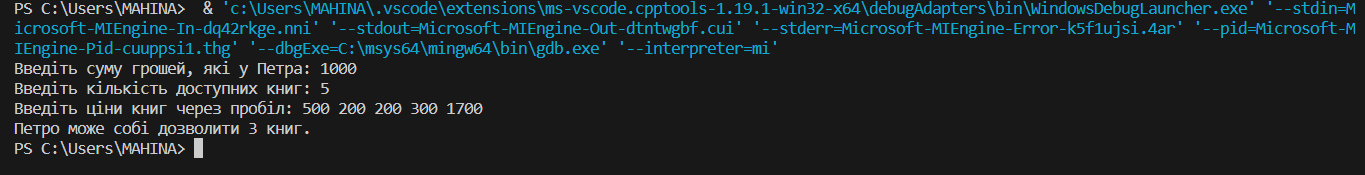
Завдання №5 Class Practice Work

Спочатку добавив три книги, попросив їхній список, взяв книжку під назвою C++, повернув цю книжку, знову попросив список книг, щоб перевірити чи вона повернулася.



*Image 21. Результати виконання завдання №6 Class Practice Wor**k*

Завдання №6 Self Practice Work

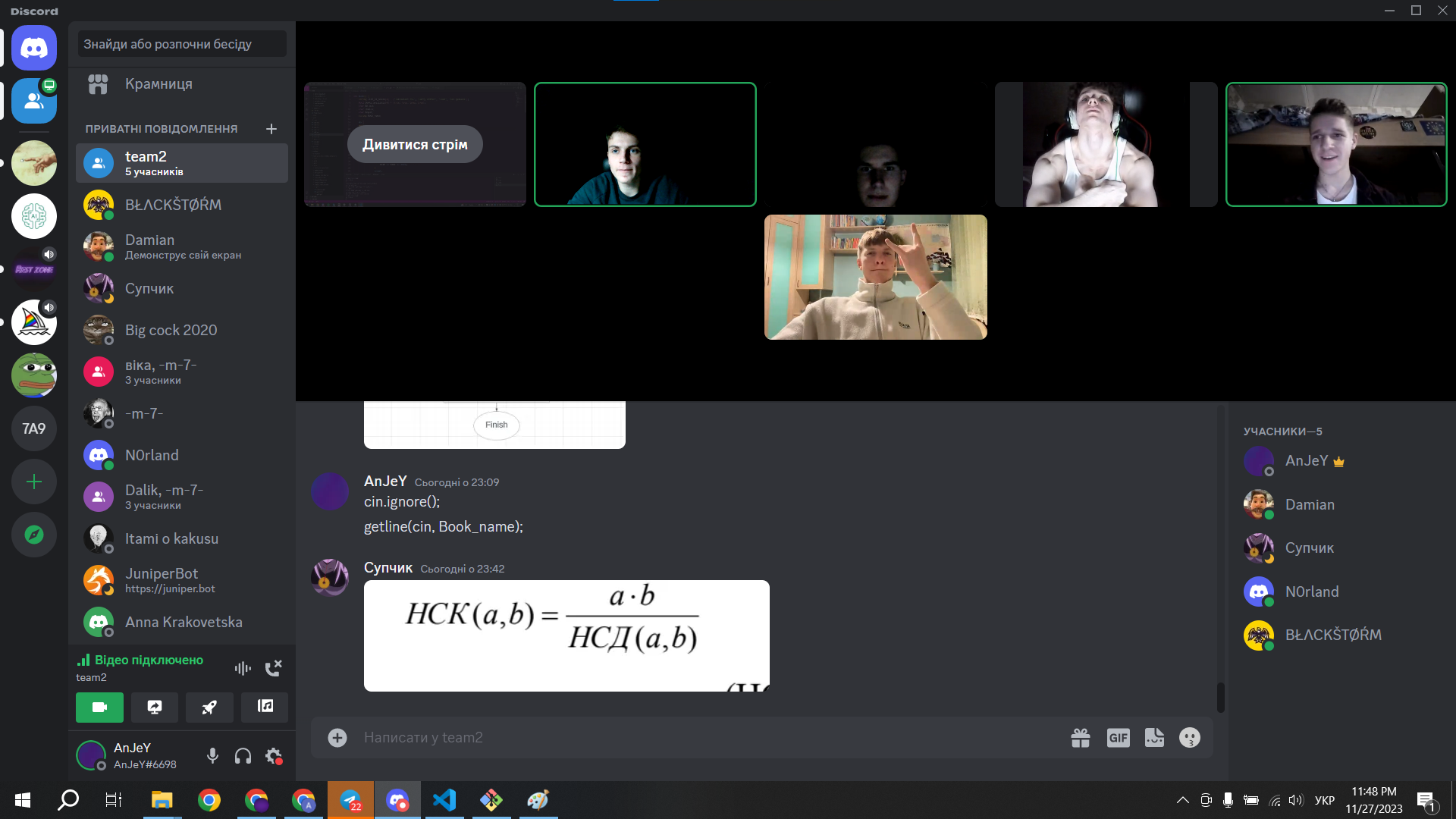


*Image 23. Результати виконання завдання №6 Self Practice Work*

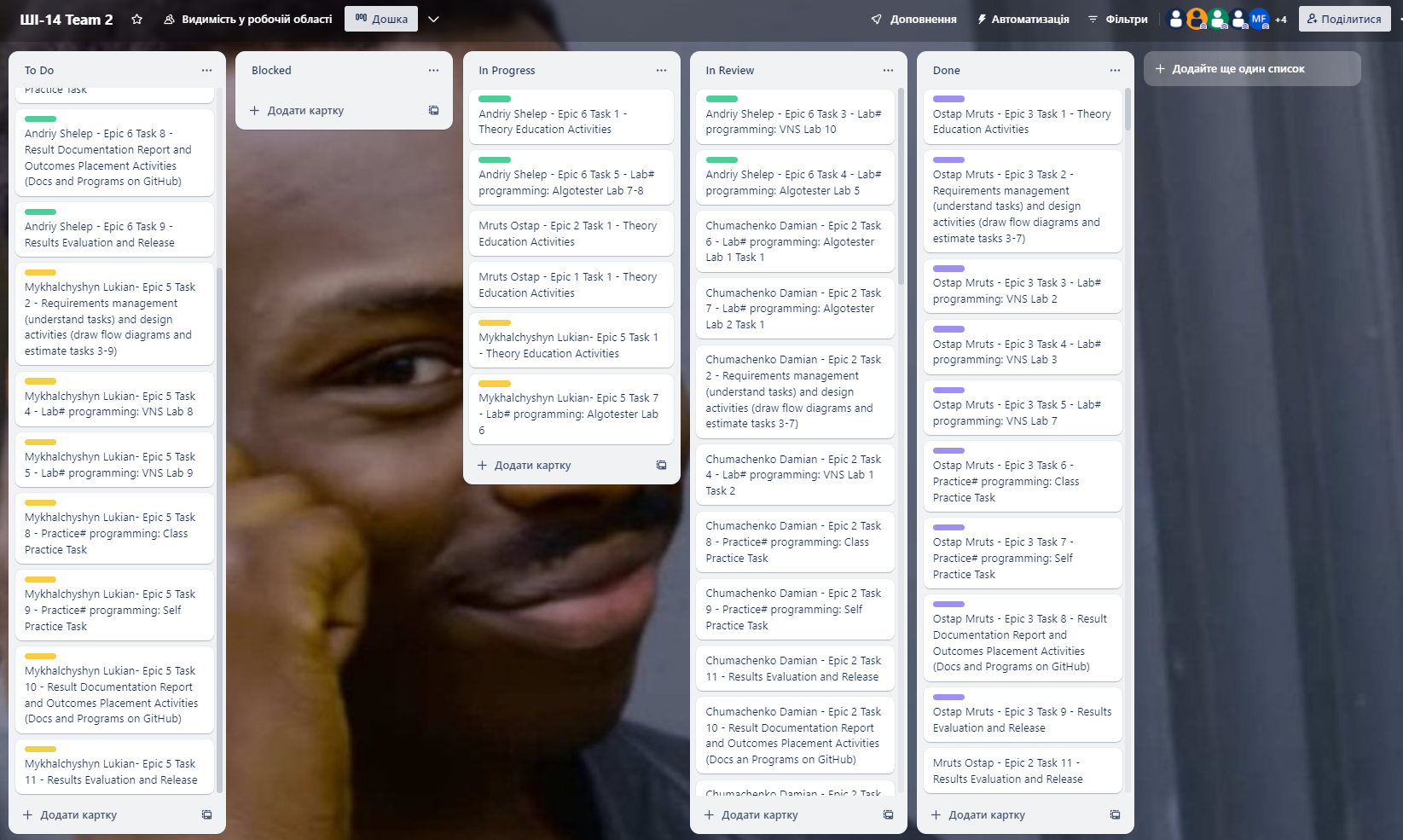
У книжковому магазині є n книг різних цін. Петро має k гривень. Напиши програму, яка визначить максимальну кількість книг, які може собі дозволити Петро на свої гроші.

## **6. Кооперація з командою:**

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



*Image 25. Кооперація з командою*

****

*Image 26. Стан дошки trello*

# **Посилання на пулреквест: https://github.com/artificial-intelligence-department/ai\_programming\_playground/pull/804**

# **Висновки:**

Опрацьовано цикли for, while, do while у С++. Ознайомився та використав функції та їхню організацію, перевантаження функції та рекурсію. Виконано лабораторні з ВНС, практичне завдання (менеджер бібліотеки).