Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра систем штучного інтелекту



**Звіт**

про виконання

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

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

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

***Виконав:***

студент групи ШІ-12

Яцишин Ігор Васильович

# **Тема роботи:**

Робота з циклами та вкладеними циклами в мові програмування C++. Функції в мові C++. Перевантажені та рекурсивні функції, функції багатьох змінних.

# **Мета роботи:**

Навчитися працювати з різними типами циклів у мові C++. Ознайомитися з поняттям функцій та навчитися застосовувати різноманітні види функцій на практиці.

# **Теоретичні відомості:**

1. Теоретичні відомості з переліком важливих тем:

* Тема №1: Цикл for.
* Тема №2: Цикли while і do while.
* Тема №3: Функції в мові C++.
* Тема №4: Перевантаження функцій.
* Тема №5: Рекурсивні функції
* Тема №6: Функції зі змінною кількістю параметрів

1. Індивідуальний план опрацювання теорії:

* Тема №1: Цикл for.
  + Джерела Інформації
    - Стаття: C++ for Loop [C++ for Loop (With Examples)](https://www.programiz.com/cpp-programming/for-loop)
  + Що опрацьовано:
    - Опрацьовано принцип роботи з циклом for, його специфікацію та основні приклади використання
    - Опрацьовано принцип побудови блок-схем для алгоритму, що включає в себе цикл for
  + Статус: Ознайомлений
  + Початок опрацювання теми: 14.11.2023 16.00
  + Звершення опрацювання теми: 14.11.2023 17.15
* Тема №2: Цикли while і do while.
  + Джерела Інформації:
    - Стаття: C++ while and do...while Loop [C++ while and do...while Loop (With Examples)](https://www.programiz.com/cpp-programming/do-while-loop)
    - Стаття: C++ do…while loop with Examples [C++ do…while loop with Examples](https://www.guru99.com/cpp-do-while-loop.html)
  + Що опрацьовано:
    - Вивчено теоретичні основи роботи з циклом while та закріплено їх на практиці
    - Опрацьовано матеріал щодо циклу do…while та його основною відмінністю від циклу while
  + Статус: Ознайомлений
  + Початок опрацювання теми: 14.11.2023 18.20
  + Звершення опрацювання теми: 14.11.2023 19.25
* Тема №3: Функції в мові C++.
  + Джерела Інформації:
    - Стаття: Functions in C++ [Functions in C++ - GeeksforGeeks](https://www.geeksforgeeks.org/functions-in-cpp/)
  + Що опрацьовано:
    - Опрацьовано основні принципи роботи з функціями в C++ та їхніми аргументами
  + Статус: Ознайомлений
  + Початок опрацювання теми: 14.11.2023 19.30
  + Звершення опрацювання теми: 14.11.2023 20.10
* Тема №4: Перевантаження функцій.
  + Джерела Інформації:
    - Стаття: C++ Function Overloading [C++ Function Overloading (With Examples)](https://www.programiz.com/cpp-programming/function-overloading)
    - Відео: Function Overloading | C++ Tutorial [Function Overloading | C++ Tutorial](https://www.youtube.com/watch?v=A9COxBajDng&ab_channel=PortfolioCourses)
  + Що опрацьовано:
    - Опрацьовано та вивчено поняття перевантажених функцій в C++, а також закріплено знання на практиці
  + Статус: Ознайомлений
  + Початок опрацювання теми: 14.11.2023 20.30
  + Звершення опрацювання теми: 14.11.2023 21.20
* Тема №5: Рекурсивні функції.
  + Джерела Інформації:
    - Відео: C++ FUNCTIONS (2020) - What is recursion? Learn recursive functions! PROGRAMMING TUTORIAL [C++ FUNCTIONS (2020) - What is recursion? Learn recursive functions! PROGRAMMING TUTORIAL](https://www.youtube.com/watch?v=MwfvXDfaZeI&ab_channel=CodeBeauty)
    - Стаття: C++ Recursion [C++ Recursion - GeeksforGeeks](https://www.geeksforgeeks.org/cpp-recursion/)
  + Що опрацьовано:
    - Вивчено принцип роботи рекурсивних функцій, опрацьовано принципи та причини їх застосування на практичних прикладах
  + Статус: Ознайомлений
  + Початок опрацювання теми: 14.11.2023 22.00
  + Звершення опрацювання теми: 14.11.2023 22.55
* Тема №6: Функції зі змінною кількістю параметрів.
  + Джерела Інформації:
    - Відео: How to create functions with a variable number of arguments using stdarg.h | C Programming Tutorial [How to create functions with a variable number of arguments using stdarg.h | C Programming Tutorial](https://www.youtube.com/watch?v=3iX9a_l9W9Y&ab_channel=PortfolioCourses)
    - Стаття: Functions with Variable Number of Arguments in C++ [Functions with Variable Number of Arguments in C++ - Scaler Topics](https://www.scaler.com/topics/cpp/functions-with-variable-number-of-arguments-in-cpp/)
  + Що опрацьовано:
    - Опрацьовано можливості та специфіку функцій зі змінною кількістю параметрів у мові C++
    - Опрацьовано принцип роботи з бібліотекою stdarg.h
  + Статус: Ознайомлений
  + Початок опрацювання теми: 14.11.2023 23.10
  + Звершення опрацювання теми: 15.11.2023 00.25

# **Виконання роботи:**

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

Завдання №1 VNS Lab 2

* Варіант завдання: 25
* Деталі завдання : обчислити суму перших семи членів ряду
* Важливі деталі для врахування в імплементації програми:

Використання циклу for для обчислення суми кількох членів ряду

Завдання №2 VNS Lab 3

* Варіант завдання: 25
* Деталі завдання: 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

* Важливі деталі для врахування в імплементації програми

Використання вкладених у цикл for циклів for і do…while

Завдання №3 VNS Lab 7 Task 1

* Варіант завдання: 6
* Деталі завдання: Написати функцію min зі змінною кількістю параметрів, що знаходить

мінімальне із чисел типу int

* Важливі деталі для врахування в імплементації програми

Використання бібліотеки stdarg.h для написання функції зі змінною кількістю параметрів

Завдання №4 VNS Lab 7 Task 2

* Варіант завдання: 6
* Деталі завдання: Написати перевантажені функції й основну програму, що їх викликає
* Важливі деталі для врахування в імплементації програми

Використання перевантаження функції для зміни типу вхідних та вихідних даних

Завдання №5 Class Practice Work

* Варіант завдання: немає
* Деталі завдання: Написати програму для простого менеджменту бібліотеки
* Важливі деталі для врахування в імплементації програми

Використання циклів for, do…while та while та вкладених циклів

Завдання №6 Self Practice Work

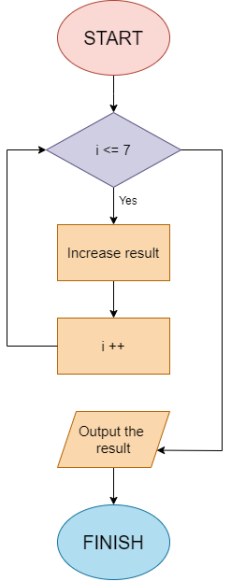
* Варіант завдання: немає
* Деталі завдання: [Зуби | Архів | Алготестер](https://algotester.com/uk/ArchiveProblem/DisplayWithEditor/20075)
* Важливі деталі для врахування в імплементації програми

Використання циклів for під час написання коду програми

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

Програма №1 VNS Lab 2

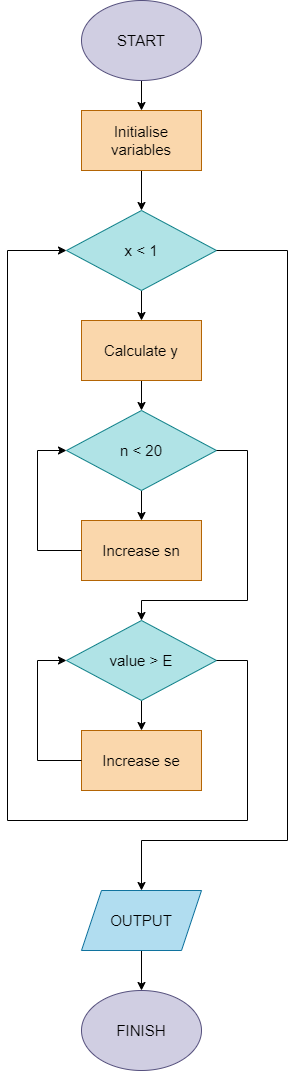
* Блок-схема



* Планований час на реалізацію: 35 хвилин

Програма №2 VNS Lab 3

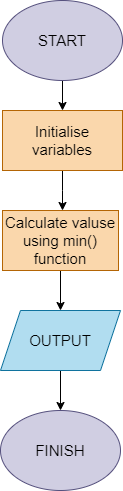
* Блок-схема



* Планований час на реалізацію:60 хвилин

Програма №3 VNS Lab 7 Task 1

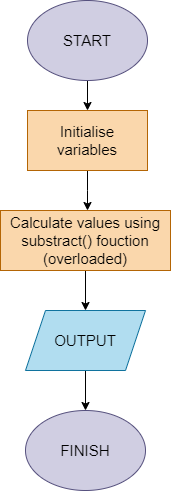
* Блок-схема



* Планований час на реалізацію: 60 хвилин

Програма №4 VNS Lab 7 Task 2

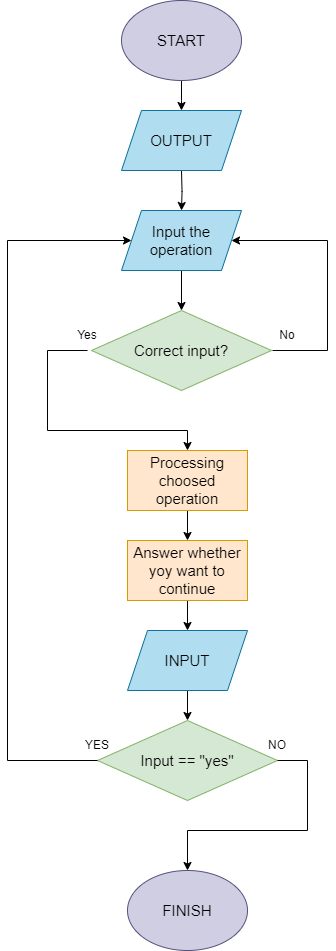
* Блок-схема



* Планований час на реалізацію: 35 хвилин

Програма №5 Class Practice Work

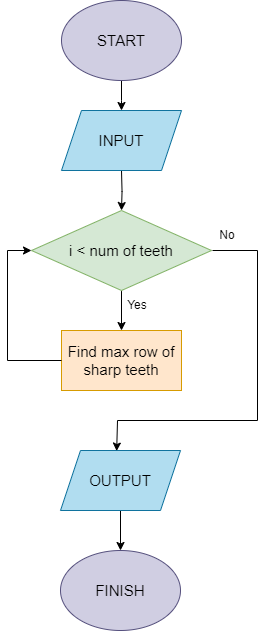
* Блок-схема



* Планований час на реалізацію: 80 хвилин

Програма №6 Self Practice Work

* Блок-схема



* Планований час на реалізацію: 40 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 2

#include <cstdio>

#include <math.h>

int main() {

double res = 0;

for(int n = 0; n <= 7; ++n){

res += pow(n, 2) \* exp(-pow(n, 0.5));

}

printf("The summ of the series equals %lf.", res);

return 0;

}

Цикл for на кожну ітерацію обчислює елемент an ,починаючи з першого і до сьомого, і додає його до результату. Кінцевий результут функцією printf виводиться в консоль

Посилання на файл програми у пул-запиті GitHub

https://github.com/artificial-intelligence-department/ai\_programming\_playground/pull/407

Завдання №2 VNS Lab 3

#include <iostream>

#include <math.h>

using namespace std;

int main () {

double x, a, b, k, eps, n, sn, se, temp, y;

n = 20;

eps = 0.0001;

k = 10;

a = 0.1;

b = 1;

sn = se = y = 0;

for(x = a; x <= b; x += (b-a)/k){

sn = x;

temp = x;

se = 0;

for(int i = 1; i <= n; i++){

sn += sn \* ((x \* x)/((2\*i) \* (2\*i + 1)));

}

int i = 1;

while(abs(temp) > eps){

se += temp;

temp = temp \* ((x \* x)/((2\*i) \* (2\*i + 1)));

++i;

}

y = (exp(x) - exp(-x)) / 2.0;

cout << "X = " << x << " SN = " << sn << " SE = " << se << " Y = " << y << endl;

}

return 0;

}

Цикл for на кожну ітерацію змінює значення x для проведення подальших обчислень вкладеними циклами а також обраховує значення функції. Цикл for обраховує суму перших 20 членів Ряду Маклорена, а цикл while обраховує цю ж суму тільки з точністю до 10^-3. Результати ітерацій виводяться у консоль

Посилання на файл програми у пул-запиті GitHub

https://github.com/artificial-intelligence-department/ai\_programming\_playground/pull/407

Завдання №3 VNS Lab 7 Task 1

#include <iostream>

#include <cstdarg>

int min(int x, ...);

int main() {

int minOf5, minOf10, minOf12;

minOf5 = min(5, 3, 17, 6, 5, 0);

minOf10 = min(10, 2, 4, 3, 17, 34, 5, 7, -4, 8, 22);

minOf12 = min(12, 3, 6, 9, 13, 22, 11, 6, 45, 1, 7, 5, 16);

std::cout << "minOf5 = " << minOf5 << std::endl;

std::cout << "minOf10 = " << minOf10 << std::endl;

std::cout << "minOf12 = " << minOf12 << std::endl;

return 0;

}

int min(int x, ...) {

va\_list args;

va\_start(args, x);

int minimalValue = va\_arg(args, int);

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

int argument = va\_arg(args, int);

if(argument < minimalValue) {

minimalValue = argument;

}

}

va\_end(args);

return minimalValue;

}

Функція min є функцією змінної кількості аргументів, яка першим аргументом приймає кількість чисел, над якими виконуватимуться дії. Результати роботи функції виводяться у консоль користувача

Посилання на файл програми у пул-запиті GitHub

https://github.com/artificial-intelligence-department/ai\_programming\_playground/pull/407

Завдання №4 VNS Lab 7 Task 2

#include <iostream>

double substract(double x, double y);

void substract(double r1, double r2, double i1, double i2, double &real, double &imaginable);

int main() {

double real, imaginable;

char op;

std::cout << substract(12.34, 7.9) << std::endl;

substract(5, 2.5, 4.3, 2.3, real, imaginable);

if(imaginable > 0) {

op = '+';

}else {

op ='-';

}

std::cout << "z = " << real << op << imaginable << "i" << std::endl;

return 0;

}

double substract(double x, double y) {

double res;

res = x - y;

return res;

}

void substract(double r1, double r2, double i1, double i2, double &real, double &imaginable) {

real = substract(r1, r2);

imaginable = substract(i1, i2);

}

Функція double substract запрограмована на віднімання дійсних чисел. В свою чергу функція void substract є перевантаженою відносно типу вихідних даних та кількості аргументів. Результати роботи обох функцій виводяться у консоль

Посилання на файл програми у пул-запиті GitHub

https://github.com/artificial-intelligence-department/ai\_programming\_playground/pull/407

Завдання №5 Class Practice Work

#include <iostream>

#include <vector>

int main() {

std::vector<std::string> books = {"Harry Potter", "The Hobbit", "Pride and Prejudice", "1984"};

std::vector<bool> availability = {true, true, true, true};

int operation, take, ret;

std::string continueWork;

while(true){

std::cout << "LIBRARY MANAGEMENT" << std::endl;

tryAgain:

std::cout << "Please choose an operation:" << std::endl;

std::cout << "1. List all books" << std::endl;

std::cout << "2. Borrow a book (if available)" << std::endl;

std::cout << "3. Return the book" << std::endl;

std::cout << "4. Exit the program" << std::endl;

std::cin >> operation;

if(operation == 1){

for(int i = 0; i < 4; i++){

std::cout << i + 1 << "." << books[i] << std::endl;

}

}else if(operation == 2){

for(auto element: availability){

if(element){

std::cout << "is avaliable \n";

}else {

std::cout << "is not avaliable \n";

}

}

std::cout << "Enter the number of book you want to borrow" << std::endl;

std::cin >> take;

if(availability[take - 1] = false) {

std::cout << "Sorry, this book has already been borrowed" << std::endl;

}else {

availability[take - 1] = false;

std::cout << "The book has been borrowed successfully" << std::endl;

}

}else if(operation == 3){

for(auto element: availability){

if(element){

std::cout << "is in library \n";

}else {

std::cout << "must be returned \n";

}

}

std::cout << "Enter the number of book you want to return" << std::endl;

std::cin >> ret;

if(availability[ret - 1] = true) {

std::cout << "This book is already in library" << std::endl;

}else{

availability[ret - 1] = true;

std::cout << "Thanks for returning the book" << std::endl;

}

}else if(operation == 4){

break;

}else{

std::cout << "Sorry, such operation doesn't exist" << std::endl;

goto tryAgain;

}

do{

std::cout << "Do you want to continue? (Enter 'yes' or 'no')" << std::endl;

std::cin >> continueWork;

if(continueWork == "no"){

return 0;

}else{

break;

}

}while (continueWork == "yes");

}

return 0;

}

У програмі використовуються цикли for, do…while та конструкція goto для реалізації роботи користувача з умовним менеджментом бібліотеки з функціями перерахування книг, можливості позичити чи повернути книгу. Робота з “бібліотекою” відбувається через консоль.

Посилання на файл програми у пул-запиті GitHub

https://github.com/artificial-intelligence-department/ai\_programming\_playground/pull/407

Завдання №6 Self Practice Work

#include <iostream>

int main() {

long long n, k, maxs = 0, currents = 0;

std::cin >> n >> k;

long long a[n];

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

std::cin >> a[i];

}

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

if(a[i] >= k){

currents++;

maxs = std::max(maxs, currents);

}else{

currents = 0;

}

}

std::cout << maxs;

return 0;

}

У програмі використовуються цикли for для введення елементів масиву та для визначення найдовшої послідовності елементів, що задовольняють певну умову (гострота зубів дрвкона)

Посилання на файл програми у пул-запиті GitHub

https://github.com/artificial-intelligence-department/ai\_programming\_playground/pull/407

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

Завдання №1 VNS Lab 2

OUTPUT:

The summ of the series equals 14.354754.

Час затрачений на виконання завдання: 30 хвилин

Завдання №2 VNS Lab 3

OUTPUT:

X = 0.1 SN = 0.100295 SE = 0.100167 Y = 0.100167

X = 0.19 SN = 0.192029 SE = 0.191143 Y = 0.191145

X = 0.28 SN = 0.28652 SE = 0.283659 Y = 0.283673

X = 0.37 SN = 0.385127 SE = 0.378442 Y = 0.3785

X = 0.46 SN = 0.489274 SE = 0.476394 Y = 0.476395

X = 0.55 SN = 0.600469 SE = 0.578149 Y = 0.578152

X = 0.64 SN = 0.720327 SE = 0.684585 Y = 0.684594

X = 0.73 SN = 0.850598 SE = 0.796564 Y = 0.796586

X = 0.82 SN = 0.993188 SE = 0.914984 Y = 0.915034

X = 0.91 SN = 1.1502 SE = 1.0409 Y = 1.0409

X = 1 SN = 1.32394 SE = 1.1752 Y = 1.1752

Час затрачений на виконання завдання: 50 хвилин

Завдання №3 VNS Lab 7 Task 1

OUTPUT:

minOf5 = -1343220080

minOf10 = -1343220711

minOf12 = 1

Час затрачений на виконання завдання: 55 хвилин

Завдання №4 VNS Lab 7 Task 2

OUTPUT:

4.44

z = 2.5+2i

Час затрачений на виконання завдання: 45 хвилин

Завдання №5 Class Practice Work

INPUT/OUTPUT(Example):

LIBRARY MANAGEMENT

Please choose an operation:

1. List all books

2. Borrow a book (if available)

3. Return the book

4. Exit the program

1

1.Harry Potter

2.The Hobbit

3.Pride and Prejudice

4.1984

Do you want to continue? (Enter 'yes' or 'no')

yes

LIBRARY MANAGEMENT

Please choose an operation:

1. List all books

2. Borrow a book (if available)

3. Return the book

4. Exit the program

2

is avaliable

is avaliable

is avaliable

is avaliable

Enter the number of book you want to borrow

1

The book has been borrowed successfully

Do you want to continue? (Enter 'yes' or 'no')

no

Час затрачений на виконання завдання: 100 хвилин

Завдання №6 Self Practice Work

INPUT:

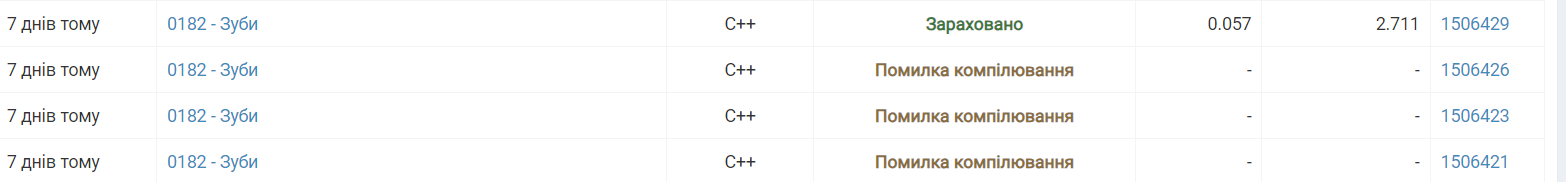
7 4

7 1 4 7 6 3 4

OUTPUT:

3

Час затрачений на виконання завдання: 40 хвилин



# **Висновки:**

Під час опрацювання теоретичного матеріалу та роботи над завданнями розділу я навчився працювати з циклами та функціями в мові C++. Виникли труднощі під час опрацювання матеріалу теми “Функції з змінною кількістю аргументів”, проте всі прогалини було надолужено. Весь пройдений матеріал закріплено практично завдяки виконанні лабораторних та практичних робіт а також самопрактиці.