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-11

Попович Олег Васильович

# **Тема роботи:**

Побудова лінійних, рекурсивних та розгалужених алгоритмів для розв’язання задач

# **Мета роботи:**

Ознайомитися з поняттями лінійного та розгалуженого алгоритмів, навчитись будувати лінійні та розгалужені алгоритми для розв’язання задач, ознайомитись зі статичним поліморфізмом

Виконати задані лабораторні роботи.

Виконати задане практичне завдання.

# **Теоретичні відомості:**

1. Теоретичні відомості з переліком важливих тем:

* Тема №1: Умовні оператори
* Тема №2: Лінійні та розгалужені алгоритми.
* Тема №3 Рекурсія.

1. Індивідуальний план опрацювання теорії:

* Тема №1: Умовні оператори.
  + Джерела Інформації
* “The C++ Programming Language” by Bjarne Stroustrup
* “C++ For Dummies” by Stephen R. Davis
  + Що опрацьовано:
* Синтаксис і семантика операторів if/else, switch/case.
* Умовні вирази.
* Логічні оператори.
* Тернарні оператори.
* Типові помилки при використанні умовних операторів.
  + Статус: Ознайомлений
  + Початок опрацювання теми: 29.09.2023
  + Звершення опрацювання теми: 05.10.2023
* Тема №2: Лінійні та розгалужені алгоритми.
  + Джерела Інформації:
* “The C++ Programming Language” by Bjarne Stroustrup
* “C++ For Dummies” by Stephen R. Davis
  + Що опрацьовано:
* Лінійні алгоритми
* Поняття лінійного алгоритму.
* Синтаксис та семантика операторів присвоювання.
* Вирази.
* Введення та виведення даних.
* Типові помилки при реалізації лінійних алгоритмів.
* Розгалужені алгоритми
* Поняття розгалуженого алгоритму.
* Синтаксис та семантика операторів if/else, switch/case.
* Умовні вирази.
* Логічні оператори.
* Типові помилки при реалізації розгалужених алгоритмів.
* Загальні теми
* Структура алгоритму.
* Ефективність алгоритму.
* Тестування алгоритму.
  + Статус: Ознайомлений
  + Початок опрацювання теми: 29.09.2023
  + Звершення опрацювання теми: 05.10.2023
* Тема №3: Рекурсія
  + Джерела Інформації:
* “The C++ Programming Language” by Bjarne Stroustrup
* “C++ For Dummies” by Stephen R. Davis
  + Що опрацьовано:
* **Поняття рекурсії**.
* **Приклади рекурсії.**.
* **Синтаксис рекурсії**.
* **Семантика рекурсії**.
* **Переваги та недоліки рекурсії**.
  + Статус: Ознайомлений
  + Початок опрацювання теми: 29.09.2023
  + Звершення опрацювання теми: 05.10.2023

# **Виконання роботи:**

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

Завдання **№1 VNS Lab 3 Task 1**

* 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..

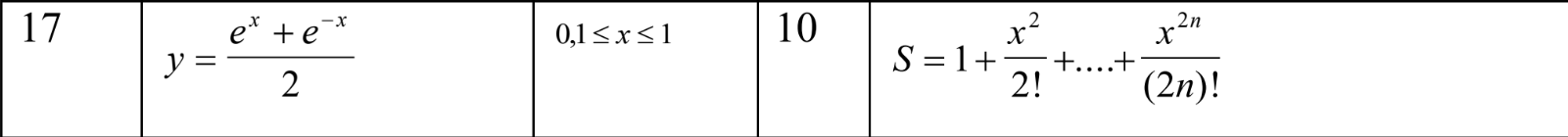


Рисунок 1 Завдання до VNS Lab 3 Task 1

Завдання **№2  VNS Lab 7 Task 1.**

* Написати функцію (або макровизначення), що знаходить площу трикутника

за його сторонами. Написати функцію perimeter зі змінною кількістю

параметрів, що знаходить периметр n-кутника за заданими сторонами. Написати

викликаючу функцію main, що звертається до функції perimeter не менше трьох

разів з кількістю параметрів 3, 5, 8.

Завдання **№3  VNS Lab 7 Task 2.**

* Написати перевантажені функції й основну програму, що їх викликає.

а) для масиву цілих чисел знаходить кількість непарних елементів;

б) для рядка знаходить кількість слів у ньому.

Завдання **№4  Practice Work**

* Програма керування бібліотекою. Книги в бібліотеці є, користувачі можуть їх взяти або повернути.
* Важливі деталі для врахування в імплементації програми

Програма повинна вміти:

1. Перерахувати всі книги.
2. Дозволити взяти книгу (за наявності).
3. Дозволити повернення книги.

Структури даних

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

Завдання **№5 Self Practice Work**

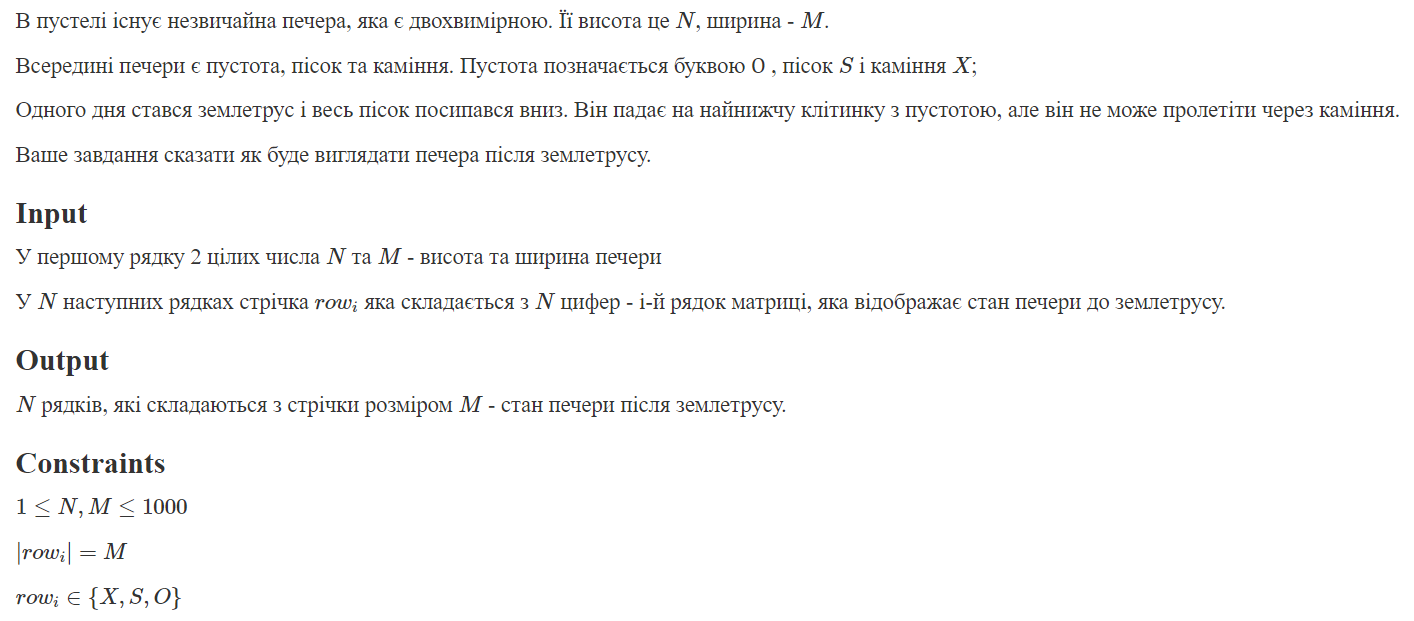


Рисунок 2 Завдання до Self Practice

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

Програма **№1 VNS Lab 3 Task 1**

* Блок-схема

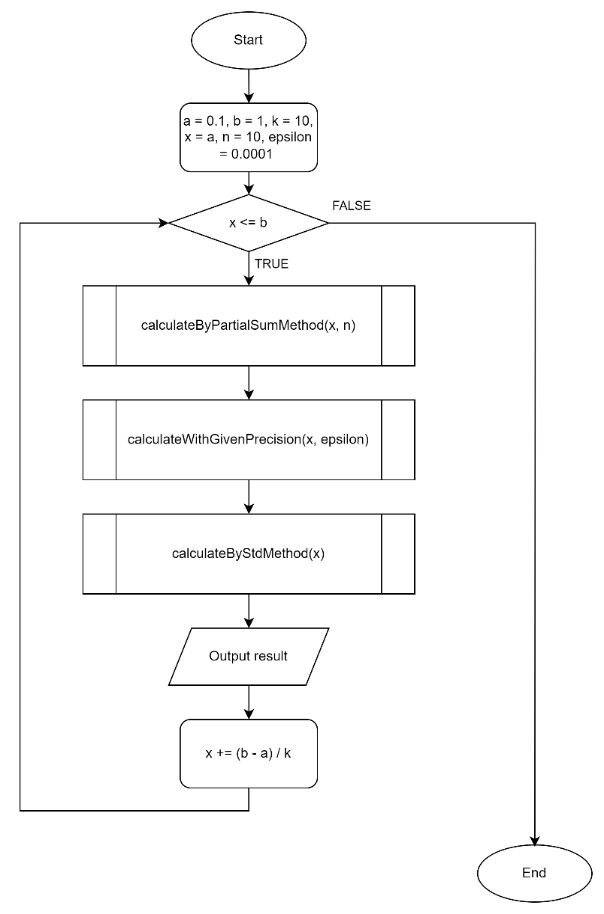


Рисунок 3 Блок схеми до програми №1

* Планований час на реалізацію: **10 хвилини**.

Програма **№2 VNS Lab 7 Task 1.**

* Блок-схема

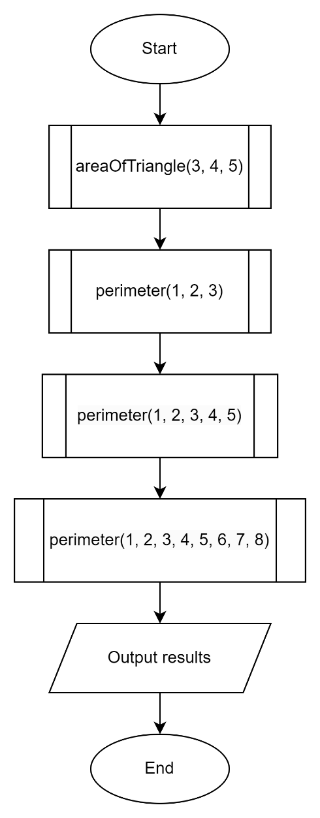


Рисунок 4 Блок схема до програми №2

* Планований час на реалізацію: **5 хвилини**.

Програма **№3 Algotester Lab 7 Task 2.**

* Блок-схема

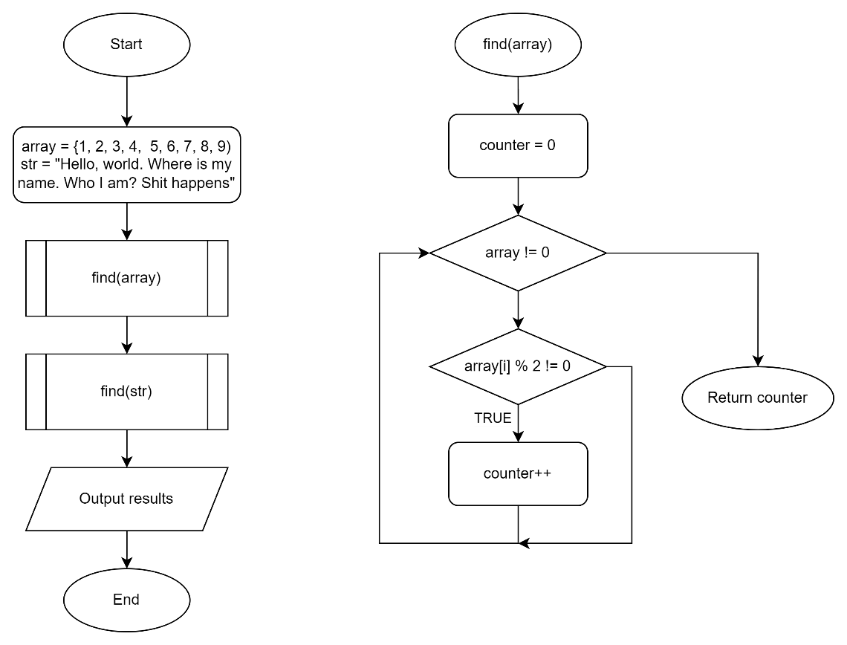


Рисунок 5 Блок схема до програми №3

* Планований час на реалізацію: **10 хвилин**.

Програма **№4 Practice Work.**

* Блок-схема

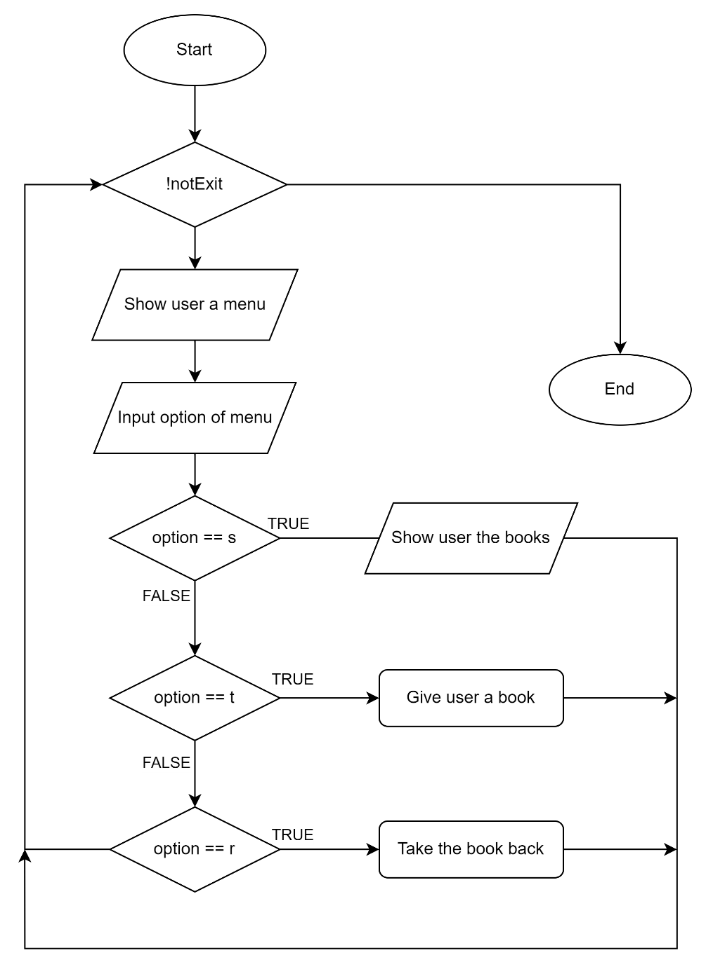


Рисунок 6 Блок схема до програми №5

* Планований час на реалізацію: **15 хвилин**.

Програма **№6 Self Practice Work.**

* Блок-схема

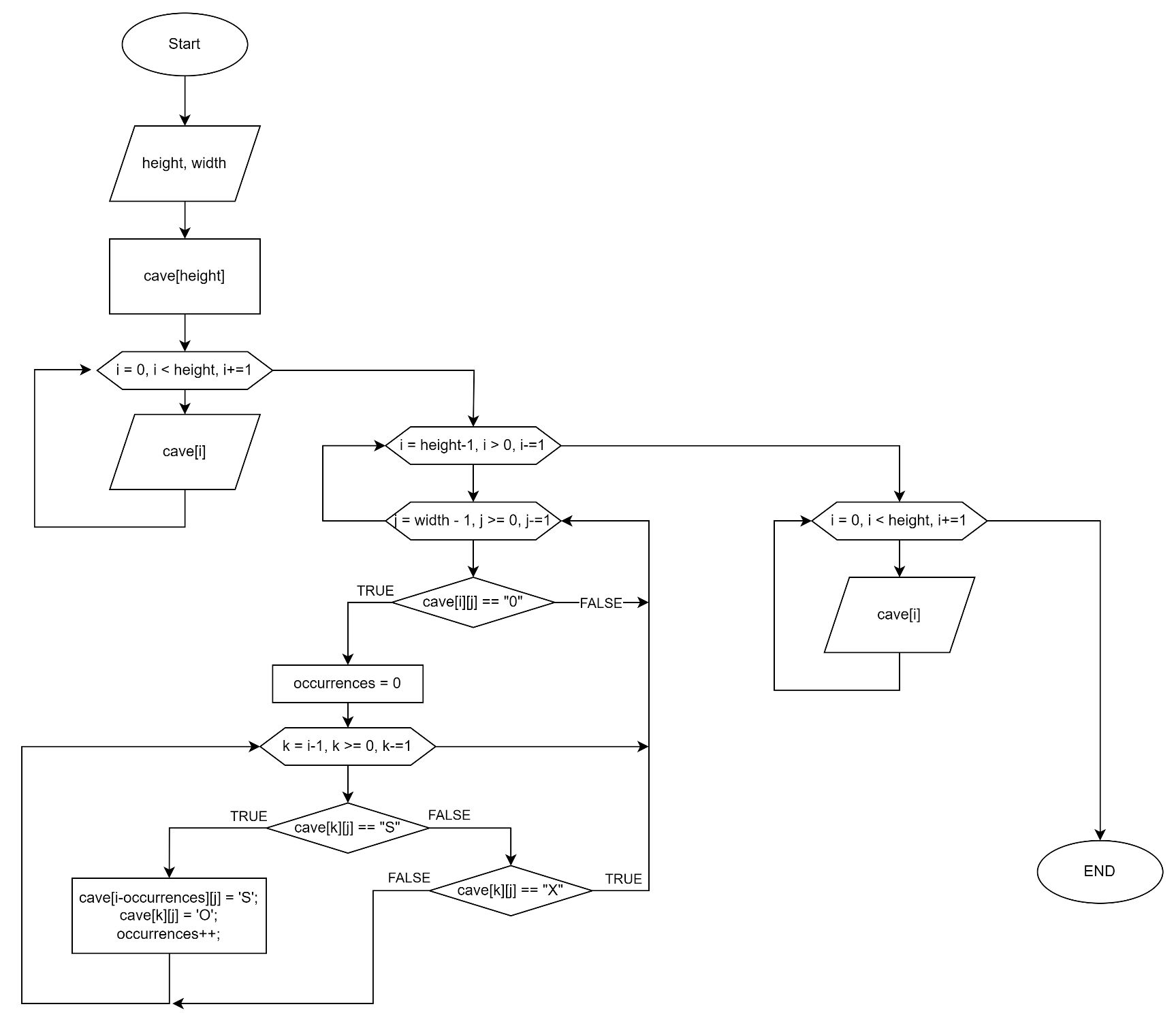


Рисунок 7 Блок схема до програми №6

* Планований час на реалізацію: **15 хвилин**.

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

Завдання **№1 VNS Lab 3 Task 1**

Деталі по програмі + Вставка з кодом з підписами до вставки

//  
// Created by olehio-h on 11/22/2023.  
//  
#include <iostream>  
#include <cmath>  
  
  
double calculateByStdMethod(double);  
double calculateByPartialSumMethod(double, int);  
double calculateWithGivenPrecision(double, double);  
  
int main(int argc, char const \*argv[])  
{  
  
 double a = 0.1, b = 1, k = 10;  
 double x = a, n = 10;  
 double epsilon = 0.0001;  
  
 while(x <= b) {  
 printf("X = %-5.2f", x);  
 printf("SN = %-10f", calculateByPartialSumMethod(x, n));  
 printf("SE = %-10f", calculateWithGivenPrecision(x, epsilon));  
 printf("Y = %6f\n", calculateByStdMethod(x));  
  
 x += (b - a) / k;  
 }  
 return 0;  
}  
  
double calculateByStdMethod(double x) {  
 return (exp(x) + exp(-x)) / 2;  
}  
  
double calculateByPartialSumMethod(double x, int n) {  
 double current = 1, sum = 1, prev;  
  
 for(int i = 0; i < n; i++) {  
 prev = current;  
 current = prev \* pow(x, 2) / ((2 \* i + 2) \* (2 \* i + 1));  
 sum += current;  
 }  
 return sum;  
}  
  
double calculateWithGivenPrecision(double x, double epsilon) {  
 double current = 1, sum = 1, prev;  
 int i = 0;  
  
 while(current >= epsilon) {  
 prev = current;  
 current = prev \* pow(x, 2) / ((2 \* i + 2) \* (2 \* i + 1));  
 sum += current;  
 i++;  
 }  
 return sum;  
}

Рисунок 8 Код до програми №1

Посилання на файл програми у пул-запиті GitHub

<https://github.com/artificial-intelligence-department/ai_programming_playground/blob/epic_3_practice_and_labs_oleh_popovych/ai_11/oleh_popovych/epic_3/vns_lab_3_task_1_oleh_popovych.cpp>

Завдання **№2  VNS Lab 7 Task 1**

Деталі по програмі + Вставка з кодом з підписами до вставки

//  
// Created by olehio-h on 11/23/2023.  
//  
#include <iostream>  
#include <cmath>  
#include <cstdarg>  
  
  
double areaOfTriangle(double, double, double);  
int perimeter(int,...);  
  
int main(int argc, char const \*argv[])  
{  
 std::cout << areaOfTriangle(3, 4, 5) << std::endl;  
 std::cout << "Perimeter is: " << perimeter(3, 3, 4, 5) << std::endl;  
 std::cout << "Perimeter is: " << perimeter(5, 3, 4, 5, 6, 7) << std::endl;  
 std::cout << "Perimeter is: " << perimeter(8, 3, 4, 5, 6, 7, 8, 9, 10) << std::endl;  
 return 0;  
}  
  
double areaOfTriangle(double a, double b, double c) {  
 if(a > b + c || b > a + c || c > a + b) {  
 return -1;  
 }  
  
 double halfOfPerimeter = (a + b + c) / 2;  
 double area = sqrt(halfOfPerimeter  
 \* (halfOfPerimeter - a)  
 \* (halfOfPerimeter - b)  
 \* (halfOfPerimeter - c));  
  
 return area;  
}  
  
int perimeter(int n,...) {  
 va\_list vargs\_ptr;  
 va\_start(vargs\_ptr, n);  
 int perimeter = 0;  
  
 while(n--) {  
 perimeter += va\_arg(vargs\_ptr, int);  
 }  
 va\_end(vargs\_ptr);  
 return perimeter;  
}

Рисунок 9 Код до програми №2

Посилання на файл програми у пул-запиті GitHub

<https://github.com/artificial-intelligence-department/ai_programming_playground/blob/epic_3_practice_and_labs_oleh_popovych/ai_11/oleh_popovych/epic_3/vns_lab_7_task_1_oleh_popovych.cpp>

Завдання **№3  Algotester Lab 7 Task q**

Деталі по програмі + Вставка з кодом з підписами до вставки

//  
// Created by olehio-p on 11/23/2023.  
//  
#include <iostream>  
#include <string>  
#include <sstream>  
  
  
int find(int\*);  
int find(const std::string&);  
  
int main(int argc, char const \*argv[])  
{  
 int arr[] {1, 2, 3, 4, 5, 6, 7, 8, 9};  
 std::cout << find(arr) << std::endl;  
 std::string str = "Hello, world. Where is my name. Who I am? Shit happens";  
 std::cout << find(str) << std::endl;  
 return 0;  
}  
  
int find(int\* ptr\_i) {  
 int counter = 0;  
  
 while(\*ptr\_i) {  
 if(\*(ptr\_i++) % 2 != 0) {  
 counter++;  
 }  
 }  
 return counter;  
}  
  
int find(const std::string& str) {  
 int counter = 0;  
 std::string token;  
 std::stringstream Stream(str);  
  
 while(std::getline(Stream, token, ' ')) {  
 counter++;  
 }  
  
 return counter;  
}

Рисунок 10 Код до програми №3

Посилання на файл програми у пул-запиті GitHub

<https://github.com/artificial-intelligence-department/ai_programming_playground/blob/epic_3_practice_and_labs_oleh_popovych/ai_11/oleh_popovych/epic_3/vns_lab_7_task_2_oleh_popovych.cpp>

Завдання **№4  Practice Work**

Деталі по програмі + Вставка з кодом з підписами до вставки

//  
// Created by olehio-p on 11/22/2023.  
//  
#include <iostream>  
#include <vector>  
  
  
int main(int argc, char const \*argv[])  
{  
 std::vector<std::string> books = **{** "Harry Potter",  
 "Lord of the Rings",  
 "Bible",  
 "Alice in the Wonderlands",  
 "Moby Dick",  
 "War and Peace",  
 "The Odyssey",  
 "The Iliad",  
 "Heart of Darkness"  
 **}**;  
 std::vector<bool> books\_availability = **{** true, true, true, true, true, true, true, true, true  
 **}**;  
 std::string menu = """1. [S]how available books\n"  
 "2. [T]ake book\n"  
 "3. [R]eturn book\n"  
 "4. [Q]uit";  
 char action = ' ', goOn = ' ', book = ' ';  
 std::string availability;  
 while(true) {  
 **home**:  
 std::cout << menu << std::endl;  
 std::cout << ": ";  
 std::cin >> action;  
 if(action == 'q')  
 break;  
 do {  
 switch (tolower(action)) {  
 case 's':  
 for (int i = 0; i < books.size(); i++) {  
 availability = books\_availability[i] ? "Available" : "Taken";  
 printf("%2d. %-25s", i + 1, books[i].c\_str());  
 printf(" - %s\n", availability.c\_str());  
 }  
 break;  
 case 't':  
 std::cout << "Enter the book number you want to take or [q] to exit: ";  
 std::cin >> book;  
 while (book != 'q') {  
 if (!books\_availability[book - '0' - 1]) {  
 std::cout << "Sorry, but the book is unavailable." << std::endl;  
 std::cout << "Please enter another book number or [q] to exit: ";  
 std::cin >> book;  
 } else {  
 books\_availability[book - '0' - 1] = false;  
 break;  
 }  
 }  
 break;  
 case 'r':  
 std::cout << "Enter the book number you want to return or [q] to exit: ";  
 std::cin >> book;  
 while (book != 'q') {  
 if (books\_availability[book - '0' - 1]) {  
 std::cout << "Sorry, but the book is already returned." << std::endl;  
 std::cout << "Please enter another book number or [q] to exit: ";  
 std::cin >> book;  
 } else {  
 books\_availability[book - '0' - 1] = true;  
 break;  
 }  
 }  
 break;  
 default:  
 std::cout << "Invalid operation." << std::endl;  
 goto **home**;  
 }  
 do {  
 std::cout << "Do you want to execute another operation [y]/[n]: ";  
 std::cin >> goOn;  
 } while((action == 's' && goOn == 'n') || (goOn != 'y' && goOn != 'n'));  
 } while(goOn == 'n');  
 }  
 return 0;  
}

Рисунок 11 Код до програми №5

Посилання на файл програми у пул-запиті GitHub

<https://github.com/artificial-intelligence-department/ai_programming_playground/blob/epic_3_practice_and_labs_oleh_popovych/ai_11/oleh_popovych/epic_3/practice_work_team_task_oleh_popovych.cpp>

Завдання **№5  Self Practice Work**

Деталі по програмі + Вставка з кодом з підписами до вставки

//  
// Created by olehio-p on 11/23/2023.  
//  
#include <iostream>  
  
  
int main(int argc, char const \*argv[])  
{  
 int height, width;  
 std::cin >> height >> width;  
  
 std::string cave[height];  
 for(int i = 0; i < height; i++) {  
 std::cin >> cave[i];  
 }  
  
 int occurrences;  
 for (int i = height - 1; i > 0; i--) {  
 for (int j = width - 1; j >= 0; j--) {  
 if (cave[i][j] == 'O') {  
 occurrences = 0;  
 for (int k = i - 1; k >= 0; k--) {  
 if (cave[k][j] == 'S') {  
 cave[i-occurrences][j] = 'S';  
 cave[k][j] = 'O';  
 occurrences++;  
 } else if(cave[k][j] == 'X') {  
 break;  
 }  
 }  
 }  
 }  
 }  
  
 for(int i = 0; i < height; i++) {  
 std::cout << cave[i] << std::endl;  
 }  
 return 0;  
}

Рисунок 12 Код до програми № 6

Посилання на файл програми у пул-запиті GitHub

## <https://github.com/artificial-intelligence-department/ai_programming_playground/blob/epic_3_practice_and_labs_oleh_popovych/ai_11/oleh_popovych/epic_3/self_practice_work_task_oleh_popovych.cpp>

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

Завдання **№1 VNS Lab 3 Task 1**

Деталі виконання і тестуванню програми

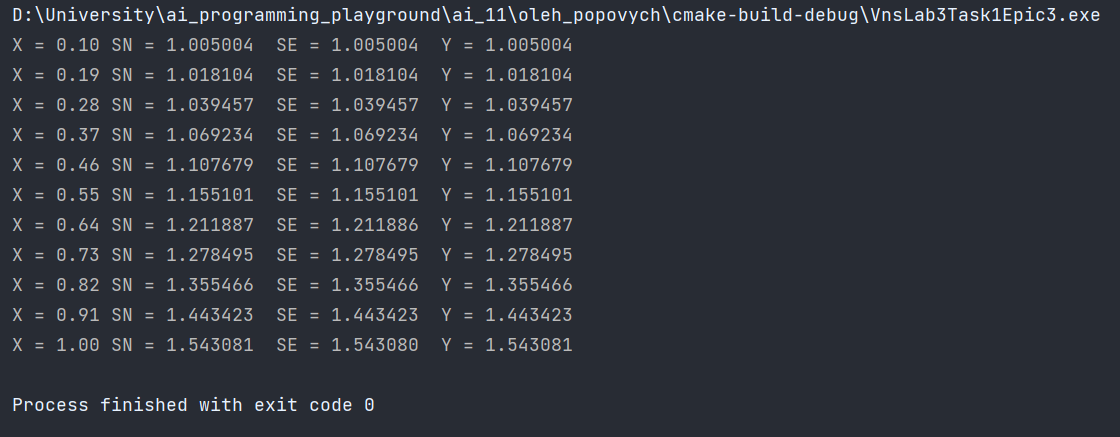


Рисунок 13 Результат виконання програми №1

Час затрачений на виконання завдання: **10 хвилин**.

Завдання **№2  VNS Lab 7 Task 1.**

Деталі виконання і тестуванню програми

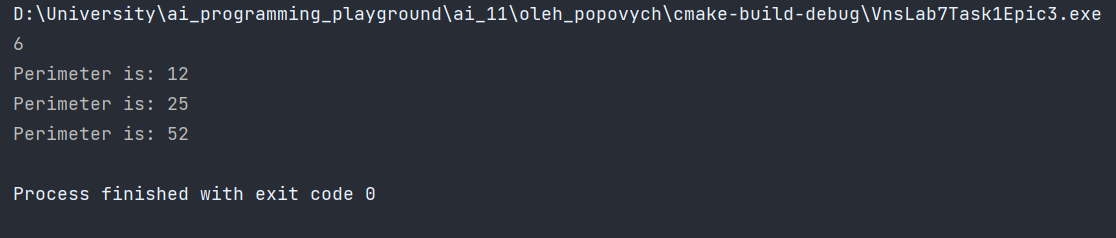


Рисунок 14 Результат виконання програми №2

Час затрачений на виконання завдання: **10 хвилини.**

Завдання **№3  VNS Lab 7 Task 2**

Деталі виконання і тестуванню програми

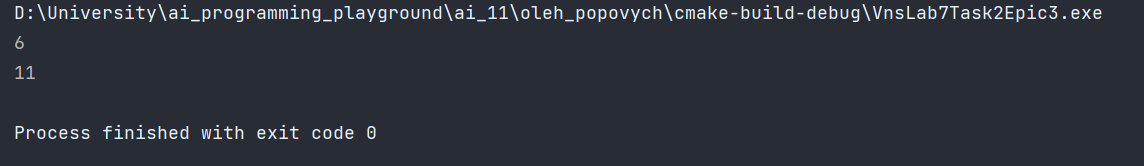


Рисунок 15 Результат виконання програми №3

Час затрачений на виконання завдання: **10 хвилин.**

Завдання **№4  Practice Work.**

Деталі виконання і тестуванню програми

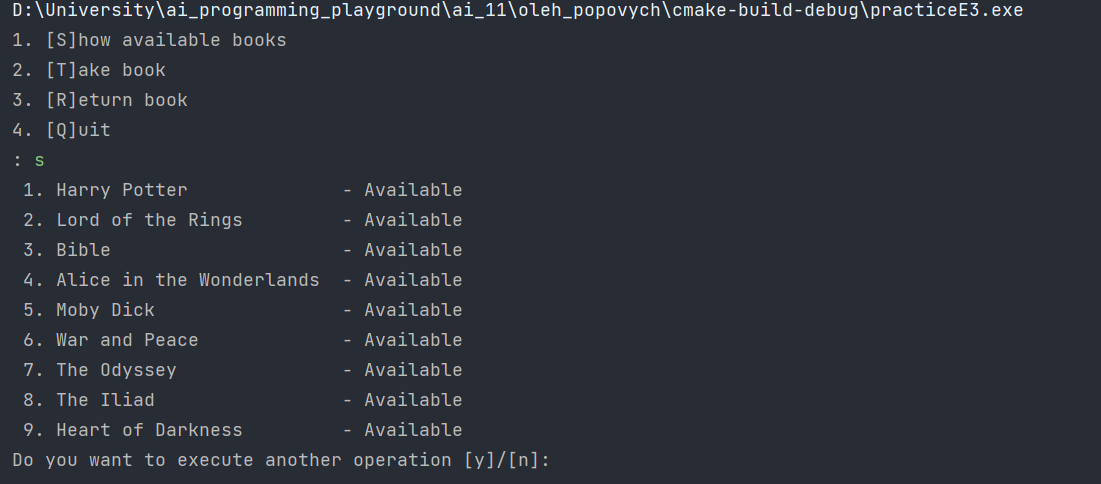


Рисунок 16 Результат виконання програми №5

Час затрачений на виконання завдання: **20 хвилин.**

Завдання **№5  Self Practice Work.**

Деталі виконання і тестуванню програми



Рисунок 17 Результат виконання програми №6

Час затрачений на виконання завдання: **15 хвилин.**

# **Висновки:**

Вивчив умовні оператори, їхні види, та використання. Лінійні та розгалужені алгоритми мови C++. Рекурсія та рекурсивні алгоритми. Отримав навички у виборі й використанні операторів С++; ознайомився з ітераційними процесами, рекурсивними алгоритмами.

Виконав задані лабораторні роботи.

Виконав задане практичне завдання.