Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра систем штучного інтелекту



**Звіт**

**про виконання розрахунково-графічних робіт блоку № 7**

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

до:

ВНС Розрахунково-графічних робіт № 1-4

Практичних Робіт № 7

***Виконала:***

студентка групи ШІ-13

Кисців Каріна Володимирівна

# **Мета роботи:**

# Одержати практичні навички в розробці і дослідженні алгоритмів розв’язання задач.

# **Виконання роботи:**

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

Завдання №1 **VNS Practice Work - Task 1**

* Варіант №18
* Розробити лінійний алгоритм для розв’язання задачі.

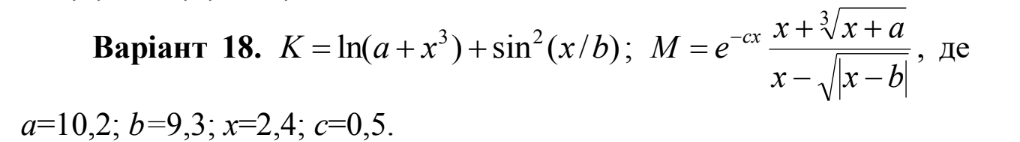


Рисунок Завдання №1 VNS Practice work - Task 1

Завдання №2 **VNS Practice Work - Task 2**

* Варіант №13
* Розробити алгоритм, що розгалужується

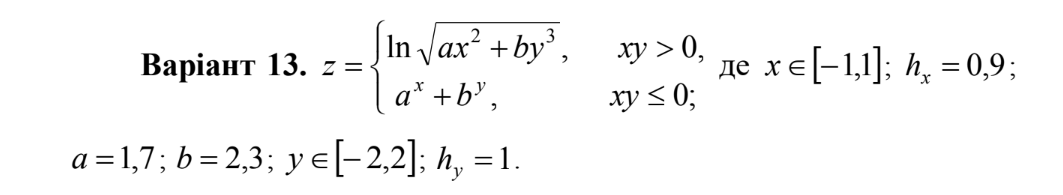


Рисунок Завдання №2 VNS Practice work -Task 2

Завдання №3 **VNS Practice Work - Task 3**

* Варіант №13

1) Обчислення об'єму циліндра

Введіть початкові дані:

Радіус основи (см) > 5.5

Висота циліндра (см) > 7

Результат - Об'єм 665.23 см.куб.

2) Обчислення опору електричного ланцюгу складається з трьох

послідовно сполучених резисторів.

Завдання №4 **VNS Practice Work - Task 4**

* Варіант №8

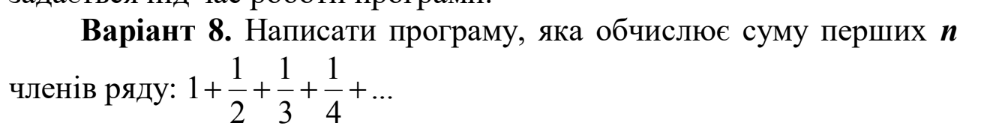


Рисунок Завдання №4 VNS Practice work - Task 4

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

Програма №1 **VNS Practice Work - Task 1**

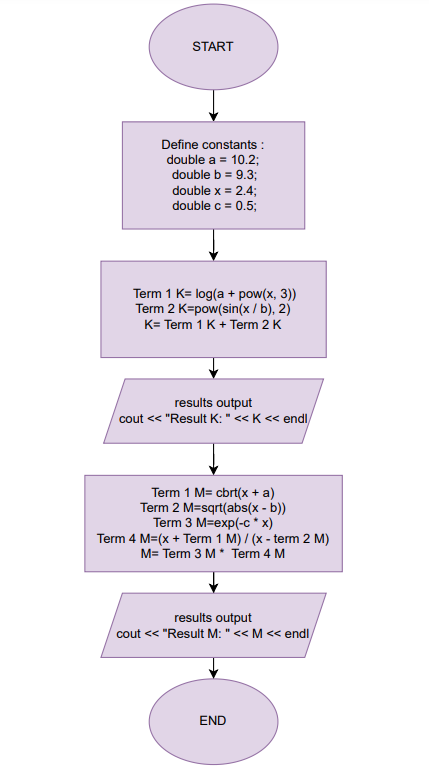


Рисунок Блок-схема до програми №1

* Планований час на реалізацію – 20 хвилин

Програма №2 **VNS Practice Work - Task 2**

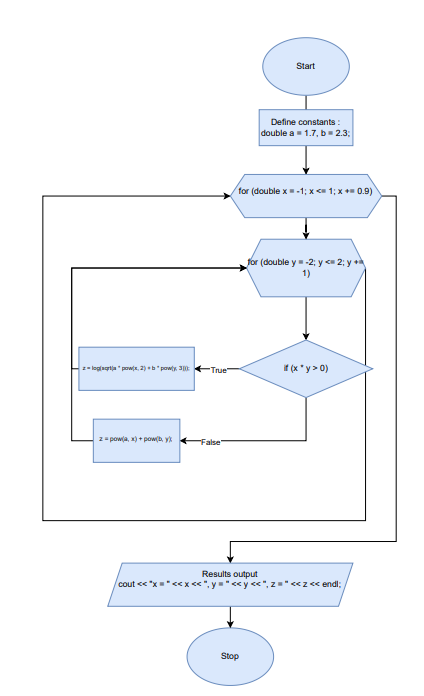
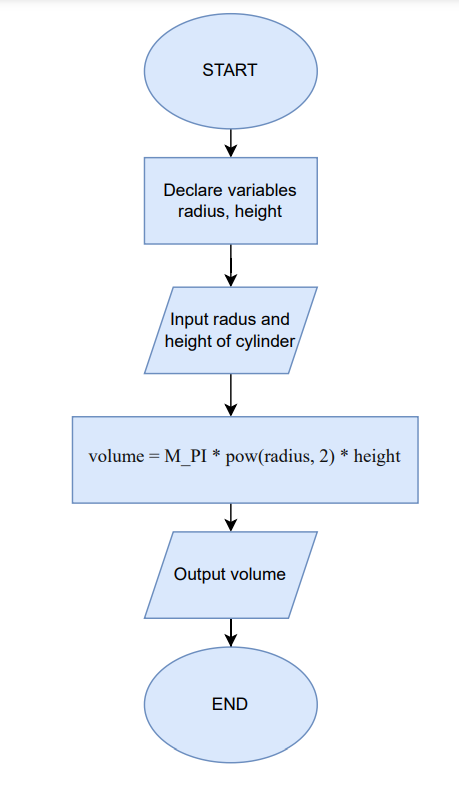
* 

Рисунок 5 Блок-схема до програми №2

Планований час на реалізацію – 20 хвилин

Програма №3.1 та 3.2 **VNS Practice Work - Task 3**



* Рисунок 6 Блок-схема до програми №3.1

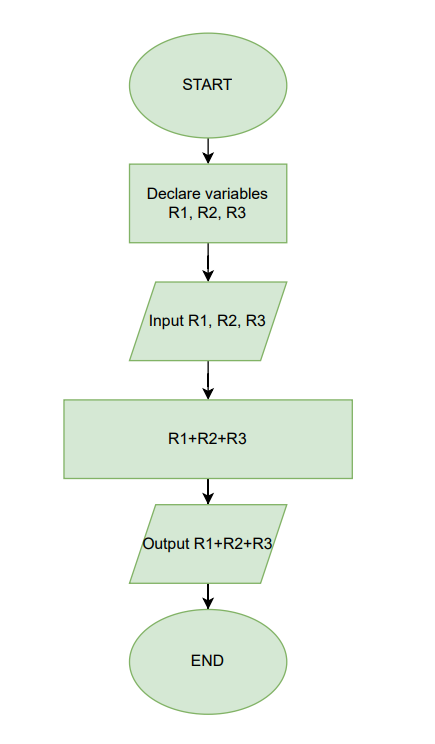


Рисунок 7 Блок-схема до програми №3.2

* Планований час на реалізацію – 40 хвилин

Програма №4 **VNS Practice Work - Task 4**

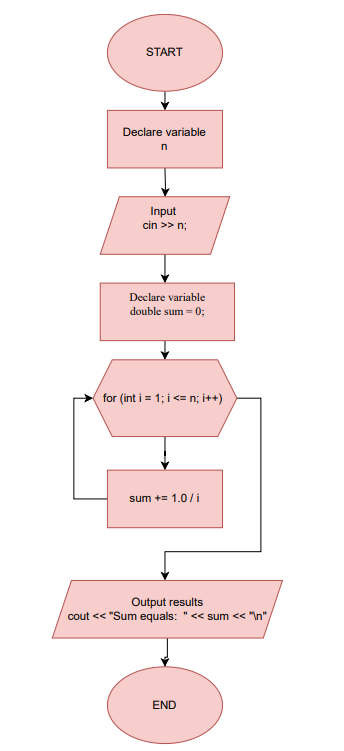


Рисунок 8 Блок-схема до програми №3.2

* Планований час на реалізацію – 20 хвилин

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

Завдання №1 **VNS Practice Work - Task 1**

[Посилання на файл програми у пул-запиті GitHub](https://github.com/artificial-intelligence-department/ai_programming_playground/pull/996/files#diff-688e7c9d532de78128b3977ba14b1296997d834174cd3ef01c851c7b437faf86)

#include <iostream>

#include <cmath>

// Функція для обчислення модулю числа

double module(double x) {

    return (x < 0) ? -x : x;

}

int main() {

    // Заданння значення константам

    double a = 10.2;

    double b = 9.3;

    double x = 2.4;

    double c = 0.5;

    // Обчислення виразу K

    double term1\_K = log(a + pow(x, 3));   // ln(a + x^3)

    double term2\_K = pow(sin(x / b), 2);   // sin^2(x/b)

    double K = term1\_K + term2\_K;

    // Виведення результату K

    std::cout << "Result K: " << K << std::endl;

    // Обчислення виразу M

    double term1\_M = cbrt(x + a);   // Кубічний корінь (x + a)

    double term2\_M = sqrt(abs(x - b));  // Квадратний корінь з модулю (x - b)

    double term3\_M = exp(-c \* x);          // e^(-cx)

    double term4\_M = (x + term1\_M) / (x - term2\_M);  // (x + cbrt(x + a)) / (x - sqrt(|x - b|))

    double M = term3\_M \* term4\_M;

    // Виведення результату M

    std::cout << "Result M: " << M << std::endl;

    return 0;

}

Рисунок 9 Програмний код до завдання №1 **VNS Practice Work - Task 1**

Завдання №2 **VNS Practice Work - Task 2**

[Посилання на файл програми у пул-запиті GitHub](https://github.com/artificial-intelligence-department/ai_programming_playground/pull/996/files#diff-5e93e9352fe6af143d24edbdf4c29d66c6396551a3efe8fc7dc7714e1db64ae1)

#include <iostream>

#include <cmath>

int main() {

    // Присвоюємо значення константам

    double const a = 1.7, b = 2.3;

    // Перебираємо значення x від -1 до 1 з кроком 0.9

    for (double x = -1; x <= 1; x += 0.9) {

        // Перебираємо значення y від -2 до 2 з кроком 1

        for (double y = -2; y <= 2; y += 1) {

            double z;

            // Перевіряємо умову x\*y > 0

            if (x \* y > 0) {

                // Обчислюємо z за формулою ln(sqrt(a\*(x^2)+b\*(y^3))

                z = log(sqrt(a \* pow(x, 2) + b \* pow(y, 3)));

            } else {

                // Обчислюємо z за формулою (a^x)+(b^y)

                z = pow(a, x) + pow(b, y);

            }

            // Виводимо результат

            std::cout << "x = " << x << ", y = " << y << ", z = " << z << std::endl;

        }

    }

    return 0;

}

Рисунок 10 Програмний код до завдання №2 **VNS Practice Work - Task 2**

Завдання №3.1 **VNS Practice Work - Task 3.1**

[Посилання на файл програми у пул-запиті GitHub](https://github.com/artificial-intelligence-department/ai_programming_playground/pull/996/files#diff-e04c9e3daf408968fb2ad5e5a53f7f7949e53b7516060863bc86c1484f53c076)

#include <iostream>

#include <cmath>

using namespace std;

int main() {

    // Введення радіуса основи

    cout << "Enter the radius of the base:";

    double radius;

    cin >> radius;

    // Введення висоти циліндра

    cout << "Enter the height of cylinder ";

    double height;

    cin >> height;

    // Обчислення об'єму циліндра

    double volume = M\_PI \* pow(radius, 2) \* height;

    // Виведення результату

    std::cout << "Volume - " << volume << " cubic centimetres \n";

    return 0;

}

Рисунок 11 Програмний код до завдання№3.1 **VNS Practice Work - Task 3.1**

Завдання №3.2 **VNS Practice Work - Task 3.2**

[Посилання на файл програми у пул-запиті GitHub](https://github.com/artificial-intelligence-department/ai_programming_playground/pull/996/files#diff-e133cf8f3463a2539c61cf4a1fa0227d10c75f760a107b9536053d4c47a0c684)

#include <iostream>

using namespace std;

int main() {

    // Оголошуємо змінні R1, R2, R3 для зберігання опорів

    int R1,R2,R3;

    // Зчитуємо значення опорів з консолі

    cin >> R1 >> R2 >> R3;

    // Виводимо суму опорів, які підключені послідовно

    cout << "Resistance of an electric circuit of three resistors connected in series" << R1+R2+R3;

    return 0;

}

Рисунок 12 Програмний код до завдання№3.2 **VNS Practice Work - Task 3.2**

Завдання №4 **VNS Practice Work - Task 4**

[Посилання на файл програми у пул-запиті GitHub](https://github.com/artificial-intelligence-department/ai_programming_playground/pull/996/files#diff-5369d6ef179f679f4304fd74f8510a42ef7a47591dde1dca3037c02123280799)

#include <iostream>

using namespace std;

int main() {

    // Оголошуємо змінну n для зберігання кількості членів ряду

    int n;

    // Зчитуємо кількість членів ряду з консолі

    cout << "Enter the number of members of the series: ";

    cin >> n;

    // Оголошуємо змінну sum для зберігання суми членів ряду

    double sum = 0;

    // Перебираємо члени ряду від 1 до n

    for (int i = 1; i <= n; i++) {

        // Додаємо до суми обернене значення поточного члена ряду

        sum += 1.0 / i;

    }

    // Виводимо суму членів ряду

    cout << "Sum equals:  " << sum << "\n";

    return 0;

}

Рисунок 13 Програмний код до завдання№4 **VNS Practice Work – Task 4**

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

Завдання №1 **VNS Practice Work - Task 1**

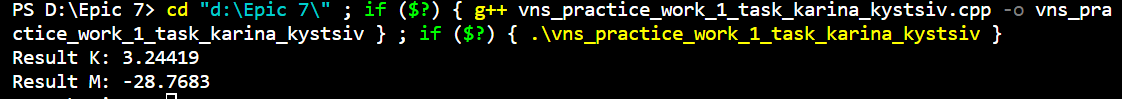


Рисунок 14 Результат виконання завдання №1 **VNS Practice Work – Task 1**

Затрачений час – 20 хвилин

Завдання №2 **VNS Practice Work - Task 2**

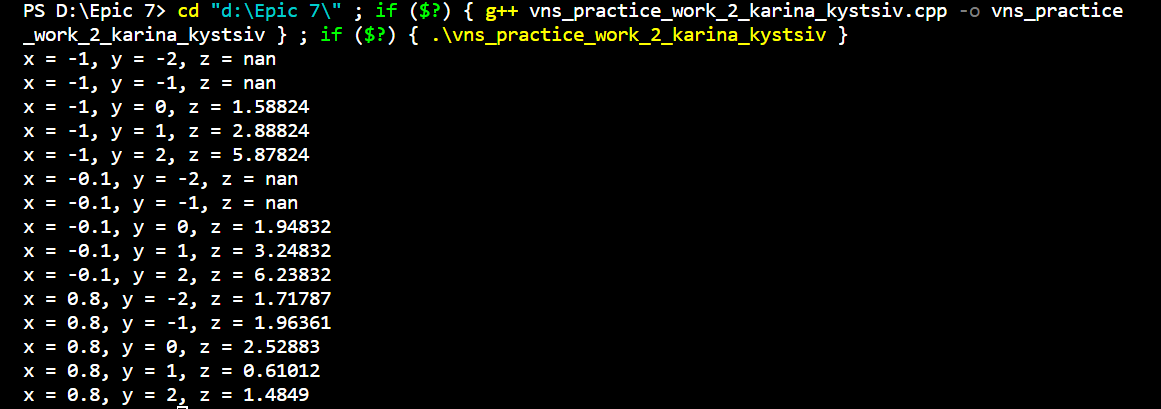


Рисунок 15 Результат виконання завдання №2 **VNS Practice Work – Task 2**

Завдання №3.1 **VNS Practice Work - Task 3.1**

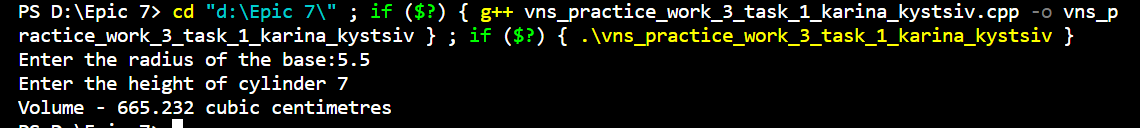


Рисунок 16 Результат виконання завдання №3.1 варіант №1 **VNS Practice Work – Task 3.1**

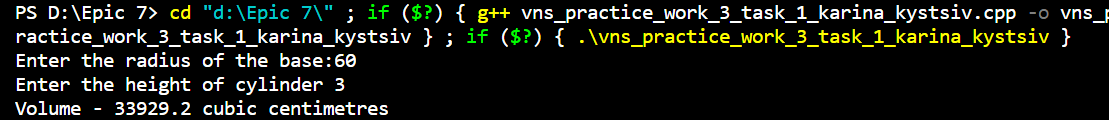


Рисунок 17 Результат виконання завдання №3.1 варіант №2 **VNS Practice Work – Task 3.1**

Завдання №3.2 **VNS Practice Work - Task 3.2**

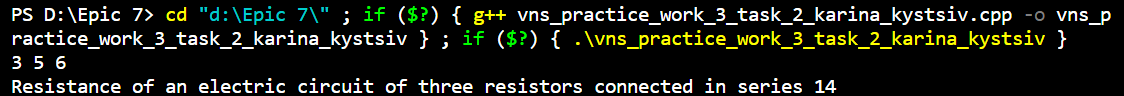


Рисунок 18 Результат виконання завдання №3.2 **VNS Practice Work – Task 3.2**

Завдання №4 **VNS Practice Work - Task 4**

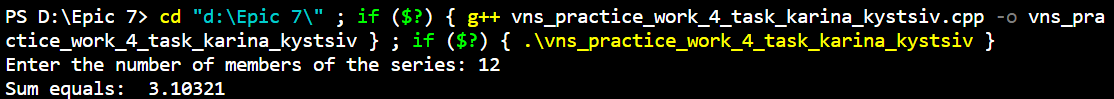


Рисунок 19 Результат виконання завдання №4 варіант №1 **VNS Practice Work – Task 4**

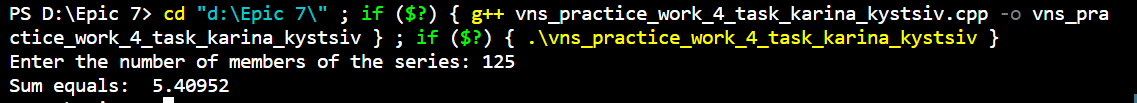


Рисунок 20 Результат виконання завдання №4 варіант №2 **VNS Practice Work – Task 4**

# **Висновки:**

Виконавши дану роботу я практично використала знання отримані впродовж семестру, розглянула різні типи алгоритмів, зокрема алгоритми з розгалуженням, різні типи циклів, вкладені цикли. Повправлялася в складанні блок-схем та написання структурованого коду.

Розрахункова робота дозволила вивчити та застосувати базові алгоритмічні конструкції мови програмування C++ для вирішення конкретної математичної задачі.