Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра систем штучного інтелекту

A blue and white logo

Description automatically generated

**Звіт**

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

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

до:

ВНС Розрахунково-графічних робіт № 1-4

Практичних Робіт № 7

**Виконав:**

Студент групи ШІ-14

Іжелюк Назарій Миколайович

**Мета роботи:** Узагальнення вивченого з курсу “Мови та парадигми програмування”.

**Виконання роботи:**

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

Завдання №1 VNS Practice 1

* **Варіант: 18**
* **Деталі завдання:**

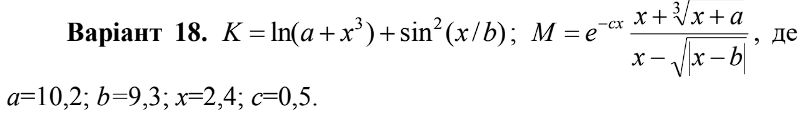
****

Рис. 1 Завдання до VNS Practice 1

* **Важливі деталі:** Розробити лінійний алгоритм для розв’язання задачі. Параметри a, b, x, c константи.

Завдання №2 VNS Practice 2

* **Варіант: 8**
* **Деталі завдання:**

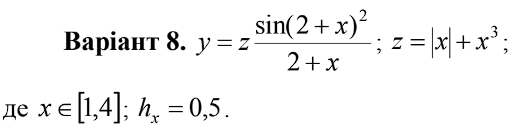
****

Рис. 2 Завдання до VNS Practice 2

* **Важливі деталі:** Розробити алгоритм, що розгалужується для розв’язання задачі номер якої відповідає порядковому номеру студента в журналі викладача

Завдання №3 VNS Practice 3

* **Варіант: 3**
* **Деталі завдання:** Обчислення об'єму куба.
* **Важливі деталі:** Написати програму згідно свого варіанту.

Завдання №4 VNS Practice 4

* **Варіант: 23**
* **Деталі завдання:**

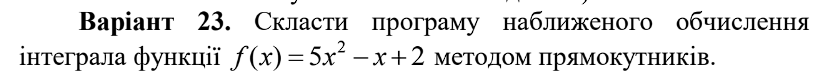
****

Рис. 3 Завдання до VNS Practice 4

* **Важливі деталі:** Написати програму згідно свого варіанту.

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

Завдання №1 VNS Practice 1

* **Блок схема:**

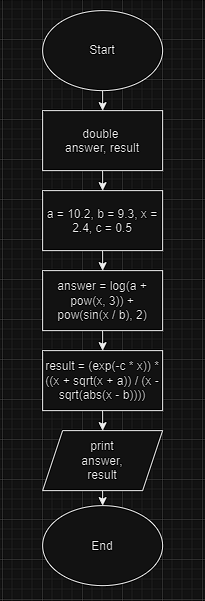
****

Рис. 4 Блок схема VNS Practice 1

Завдання №2 VNS Practice 2

* **Блок схема:**

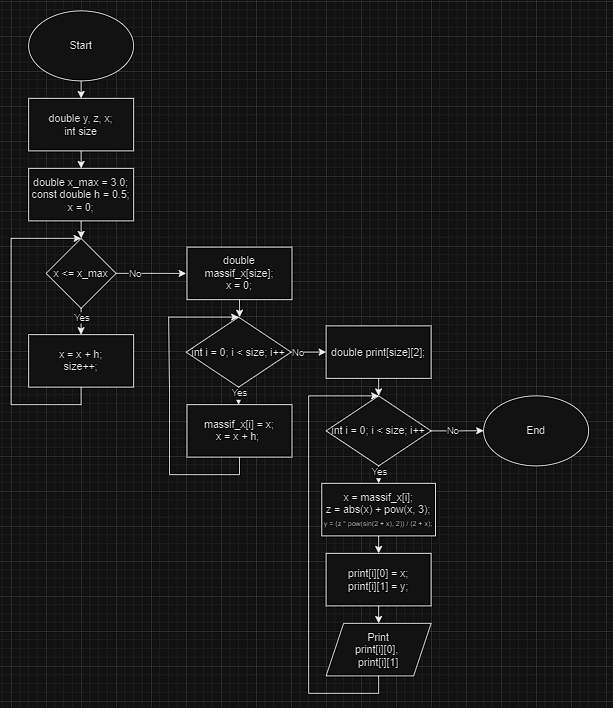
****

Рис. 5 Блок схема VNS Practice 2

Завдання №3 VNS Practice 3

* **Блок схема:**

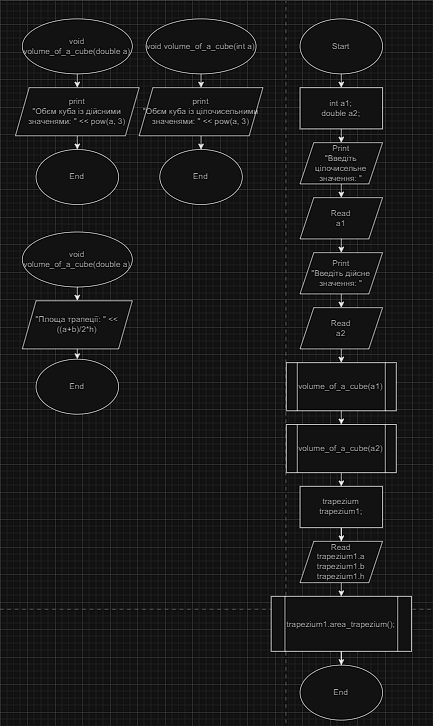
****

Рис. 6 Блок схема VNS Practice 3

Завдання №4 VNS Practice 4

* **Блок схема:**

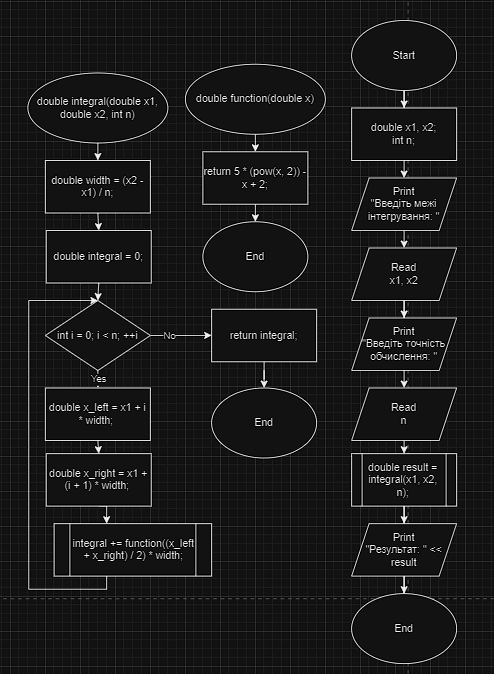
****

Рис. 7 Блок схема VNS Practice 4

## **4. Код програм:**

Код до завдання №1:

#include <iostream>

#include <math.h>

using namespace std;

int main() {

double a = 10.2, b = 9.3, x = 2.4, c = 0.5;

double answer = log(a + pow(x, 3)) + pow(sin(x / b), 2);

double result = (exp(-c \* x)) \* ((x + sqrt(x + a)) / (x - sqrt(abs(x - b))));

cout << "Результат K: " << answer << endl;

cout << "Результат M: " << result << endl;

return 0;

}

Код до завдання №2:

#include<iostream>

#include<cmath>

using namespace std;

int main(){

double y, z, x;

double x\_max = 3.0; //3. в коді використана як мінімум одна дійсний з подвійною точністю змінна

const double h = 0.5; //4. в коді використана як мінімум одна константа

int size = 0; //1. в коді використана як мінімум одна цілочисельна змінна

x = 0;

while(x <= x\_max){ //9. в коді використаний while цикл

x = x + h;

size++;

}

double massif\_x[size]; //6. в коді використаний одновимірний масив

x = 0;

for(int i = 0; i < size; i++){ //10. в коді використаний for цикл

massif\_x[i] = x;

x = x + h;

}

double print[size][2]; //7. в коді використаний двовимірний масив

cout << " | " << " x " << " | " << " y " << " | " << endl;

for(int i = 0; i < size; i++){ //10. в коді використаний for цикл

x = massif\_x[i];

z = abs(x) + pow(x, 3);

y = (z \* pow(sin(2 + x), 2)) / (2 + x);

print[i][0] = x; // Заповнюємо перший стовпець x

print[i][1] = y; // Заповнюємо другий стовпець y

cout << "-----------------" << endl;

cout << " | " << print[i][0] << " | " << print[i][1] << " | " << endl;

}

return 0;

}

Код до завдання №3:

#include<iostream>

#include<math.h>

using namespace std;

struct trapezium //14. в коді використано структури

{

double a;

double b;

double h;

void area\_trapezium(){

cout << "Площа трапеції: " << ((a+b)/2\*h);

}

};

void volume\_of\_a\_cube(int a){

cout << "Обєм куба із цілочисельними значенями: " << pow(a, 3) << endl; //17. в коді використано математичні операції та математичні функції

}

void volume\_of\_a\_cube(double a){

cout << "Обєм куба із дійсними значенями: " << pow(a, 3) << endl; //17. в коді використано математичні операції та математичні функції

}

int main(){

int a1;

double a2; //2. в коді використана як мінімум одна дійсна змінна

trapezium trapezium1;

cout << "Введіть цілочисельне значення: ";

cin >> a1;

cout << "Введіть дійсне значення: ";

cin >> a2;

volume\_of\_a\_cube(a1); //12. в коді використано перевантаження функції

volume\_of\_a\_cube(a2); //12. в коді використано перевантаження функції

cout << "Введіть 1 основу трапеції: ";

cin >> trapezium1.a;

cout << "Введіть 2 основу трапеції: ";

cin >> trapezium1.b;

cout << "Введіть висоту трапеції: ";

cin >> trapezium1.h;

trapezium1.area\_trapezium();

return 0;

}

Код до завдання №4:

#include<iostream>

#include<math.h>

using namespace std;

double function(double x) { //16. в коді використано параметри та аргументи функції

return 5 \* (pow(x, 2)) - x + 2; //17. в коді використано математичні операції та математичні функції

}

double integral(double x1, double x2, int n) { // 16. в коді використано параметри та аргументи функції

double width = (x2 - x1) / n;

double integral = 0;

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

double x\_left = x1 + i \* width;

double x\_right = x1 + (i + 1) \* width;

integral += function((x\_left + x\_right) / 2) \* width; //15. в коді використано рекурсивну функцію

}

return integral;

}

int main() {

double x1, x2; //3. в коді використана як мінімум одна дійсний з подвійною точністю змінна

int n; //1. в коді використана як мінімум одна цілочисельна змінна

cout << "Введіть межі інтегрування: ";

cin >> x1 >> x2;

cout << "Введіть точність обчислення: ";

cin >> n; //21. в коді використано оператори виведення та введення даних

double result = integral(x1, x2, n); //16. в коді використано параметри та аргументи функції

cout << "Результат: " << result << endl; //21. в коді використано оператори виведення та введення даних

return 0;

}

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

Завдання №1 VNS Practice 1



Рис. Результат тестування VNS Practice 1

Час затрачений на виконання завдання: 20 хв.

Завдання №2 VNS Practice 2

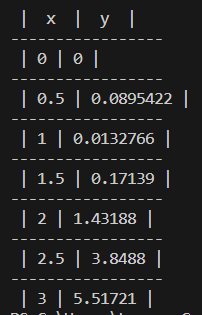


Рис. Результат тестування VNS Practice 2

Час затрачений на виконання завдання: 20 хв.

Завдання №3 VNS Practice 3

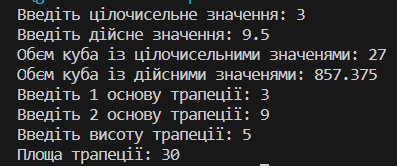


Рис. Результат тестування VNS Practice 3

Час затрачений на виконання завдання: 20 хв.

Завдання №4 VNS Practice 4

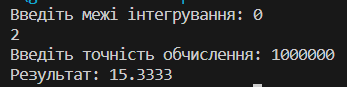


Рис. Результат тестування VNS Practice 4

Час затрачений на виконання завдання: 20 хв.

**6. Висновки:**

У процесі виконання розрахунково-графічної роботи вдалося успішно застосувати та використати ключові концепції та матеріали, вивчені під час курсу "Мови та парадигми програмування".