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:*** «Основи програмування»

до:

ВНС Розрахунково-графічних робіт № 1-4

Практичних Робіт до блоку № 7

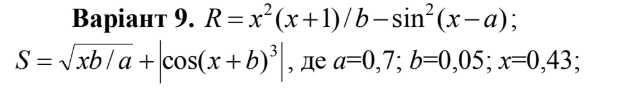
**Виконав:**

Студент групи ШІ-12

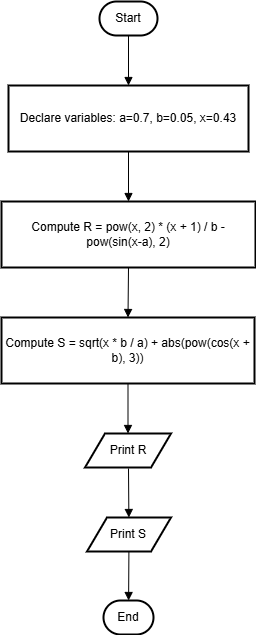
Михальчук Антон

**Завдання 1**

Розробити лінійний алгоритм для розв’язання задачі.



**Блок-схема:**



**Код:**

#include <iostream>

#include <cmath>

using namespace std;

int main() {

    double a = 0.7, b = 0.05, x = 0.43;

    double R, S;

    R = pow(x, 2) \* (x + 1) / b - pow(sin(x-a), 2);

    S = sqrt(x \* b / a) + abs(pow(cos(x + b), 3));

    cout << "R = " << R << endl;

    cout << "S = " << S << endl;

    return 0;

}

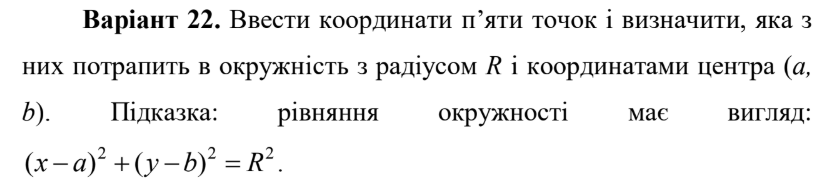
**Вивід:**



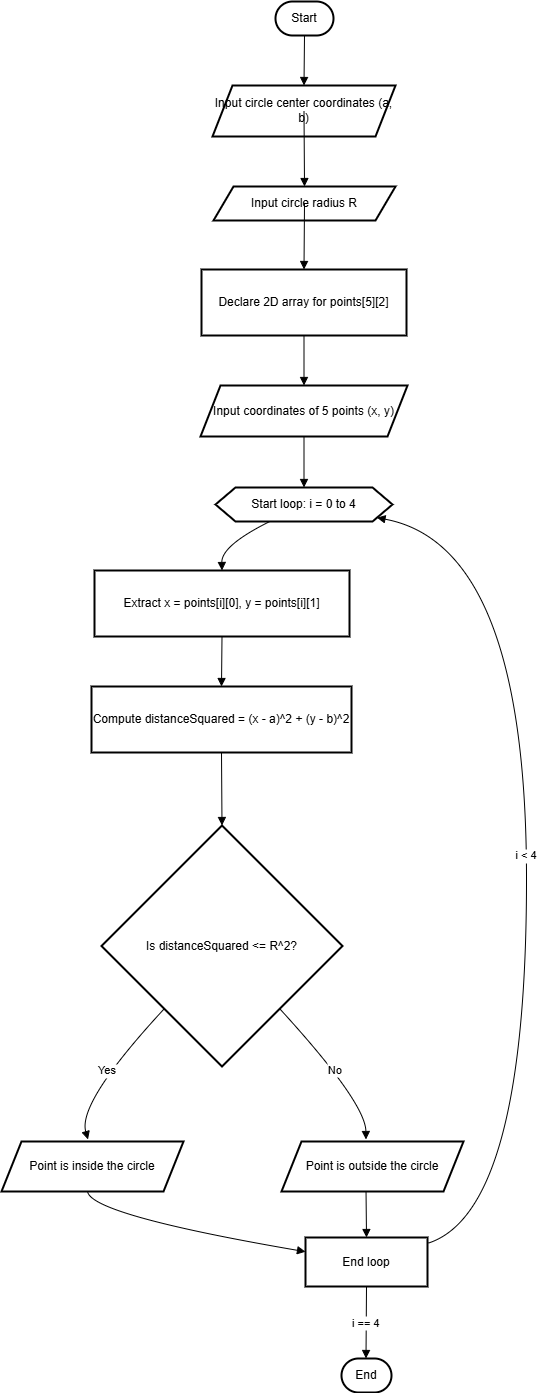
**Завдання 2.** Розробити алгоритм, що розгалужується для

розв’язання задачі номер якої відповідає порядковому номеру

студента в журналі викладача



**Блок-схема:**



**Код:**

#include <iostream>

#include <cmath>

int main() {

    double a, b, R;

    std::cout << "Enter the circle center coordinates (a, b): ";

    std::cin >> a >> b;

    std::cout << "Enter the circle radius R: ";

    std::cin >> R;

    double points[5][2];

    std::cout << "Enter the coordinates of 5 points (x, y):\n";

    for (int i = 0; i < 5; ++i) {

        std::cout << "Point " << i + 1 << " (x, y): ";

        std::cin >> points[i][0] >> points[i][1];

    }

    std::cout << "\nResults:\n";

    for (int i = 0; i < 5; ++i) {

        double x = points[i][0];

        double y = points[i][1];

        double distanceSquared = (x - a) \* (x - a) + (y - b) \* (y - b);

        if (distanceSquared <= R \* R) {

            std::cout << "Point " << i + 1 << " (" << x << ", " << y << ") is inside the circle.\n";

        } else {

            std::cout << "Point " << i + 1 << " (" << x << ", " << y << ") is outside the circle.\n";

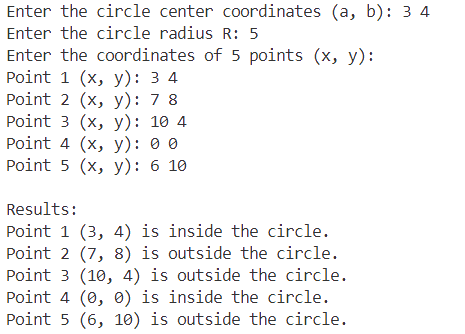
        }

    }

    return 0;

}

**Вивід:**



**Завдання 3.** Написати программу згідно свого варіанту.

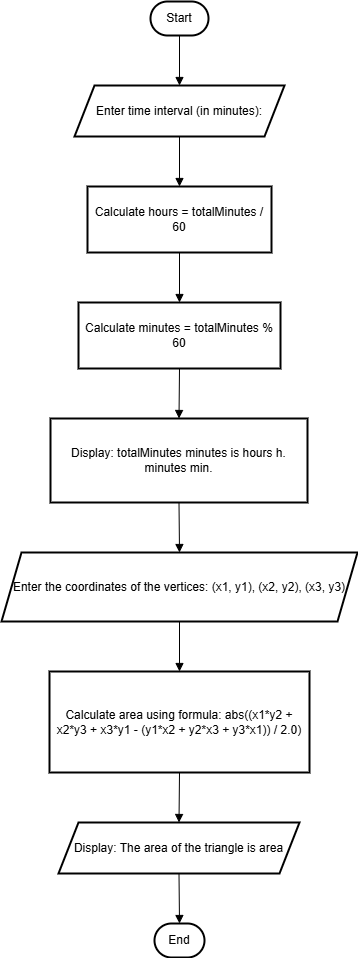
**Варіант 17.** Перерахунок величини тимчасового інтервалу, заданого в хвилинах, у величину, виражену в годинах і хвилинах. Нижче приведений вид екрану під час виконання програми, що рекомендується (дані, які вводяться користувачем, виділені напівжирним шрифтом).

Введіть часовий інтервал (в хвилинах) > 150

150 хвилин - це 2 ч. 30 мін.

Реалізувати обчислення площі трикутника, якщо відомі координати його кутів.

**Блок-схема:**

****

**Код:**

#include <iostream>

#include <cmath>

using namespace std;

int main() {

    int totalMinutes;

    cout << "Enter time interval (in minutes) > ";

    cin >> totalMinutes;

    int hours = totalMinutes / 60;

    int minutes = totalMinutes % 60;

    cout << totalMinutes << " minutes is " << hours << " h. " << minutes << " min." << endl;

    double x1, y1, x2, y2, x3, y3;

    cout << "Enter the coordinates of the first vertex (x1, y1): ";

    cin >> x1 >> y1;

    cout << "Enter the coordinates of the second vertex (x2, y2): ";

    cin >> x2 >> y2;

    cout << "Enter the coordinates of the third vertex (x3, y3): ";

    cin >> x3 >> y3;

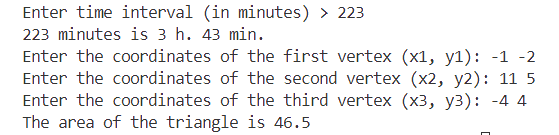
    double area = abs(((x1 \* y2 + x2 \* y3 + x3 \* y1) - (y1 \* x2 + y2 \* x3 + y3 \* x1)) / 2.0);

    cout << "The area of the triangle is " << area << endl;

    return 0;

}

**Вивід:**

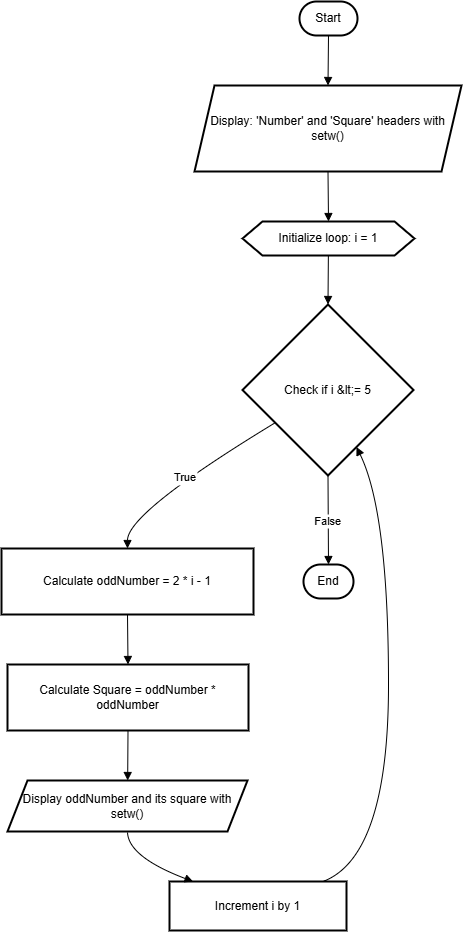


**Завдання 4.** Написати програму згідно свого варіанту.

**Варіант 4.** Написати програму, яка виводить таблицю

квадратів перших п'яти цілих позитивних непарних чисел.

**Блок-схема:**



**Код:**

#include <iostream>

#include <iomanip>

using namespace std;

int main() {

    cout << setw(10) << "Number" << setw(15) << "Square" << endl;

    for (int i = 1; i <= 5; ++i) {

        int oddNumber = 2 \* i - 1;

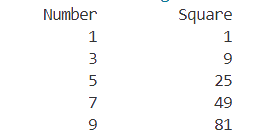
        cout << setw(10) << oddNumber << setw(15) << oddNumber \* oddNumber << endl;

    }

    return 0;

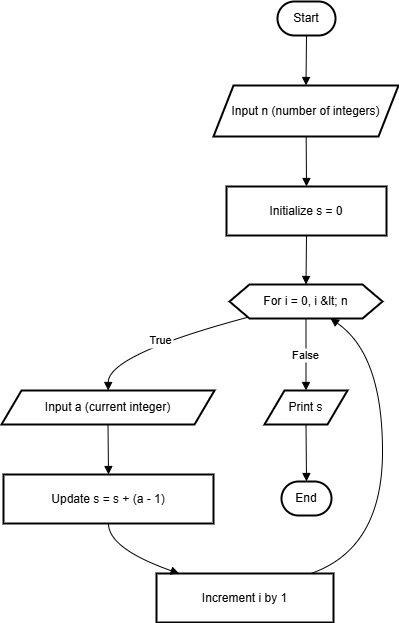
}

**Вивід:**



**Завдання з алготестеру 1. 0011 - Марічка і печиво**

**Блок-схема:**



**Код:**

#include <iostream>

using namespace std;

int main() {

    long long n, a, s = 0;

    cin >> n;

    for (int i = 0; i < n; i++) {

        cin >> a;

        s += a-1;

    }

    cout << s;

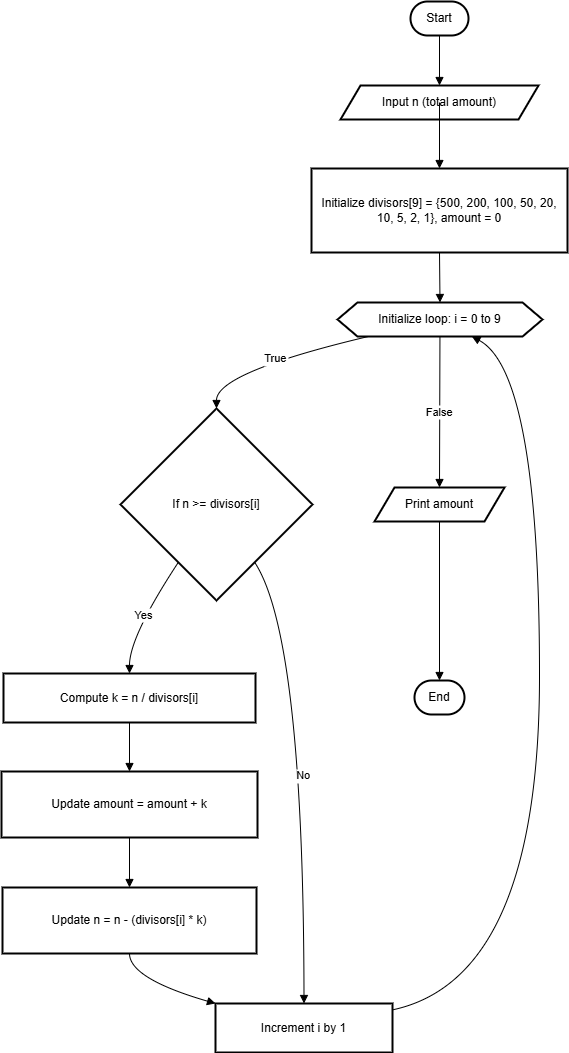
}

**Зарахування:**



**Завдання з алготестеру 2. 0021 - Депутатські гроші**

**Блок-схема:**



**Код:**

#include <iostream>

using namespace std;

int main() {

    long long n;

    cin >> n;

    int divisors[9] = {500, 200, 100, 50, 20, 10, 5, 2, 1};

    int amount = 0;

    for (int i = 0; i < 9; i++) {

        if (n >= divisors[i]) {

            int k = n / divisors[i];

            amount += k;

            n -= divisors[i] \* k;

        }

    }

    cout << amount;

    return 0;

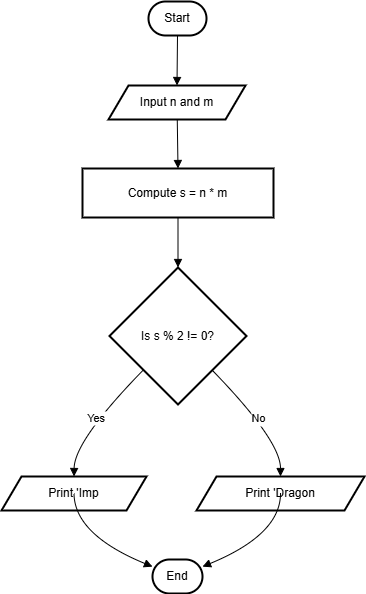
}

**Зарахування:**



**Завдання з алготестеру 3. 0181 - Цікава гра**

**Блок-схема:**



**Код:**

#include <iostream>

using namespace std;

int main() {

    int n, m, s;

    cin >> n >> m;

    s = n \* m;

    if (s % 2) {

        cout << "Imp";

    }

    else {

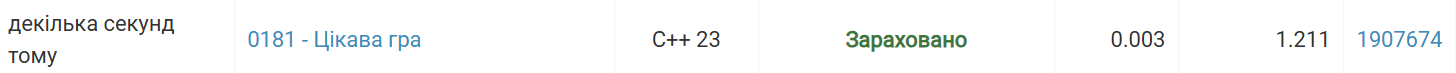
        cout << "Dragon";

    }

    return 0;

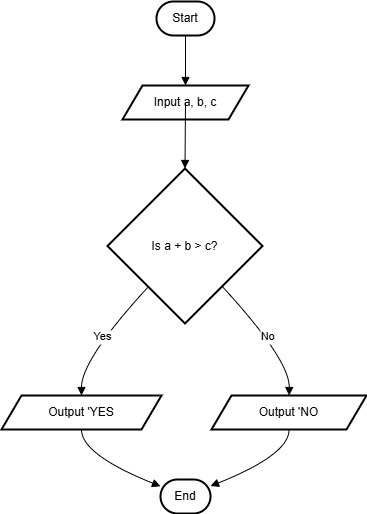
}

**Зарахування:**



**Завдання з алготестеру 4. 0011 - Марічка і печиво**

**Блок-схема:**



**Код:**

#include <iostream>

using namespace std;

int main() {

    int a, b, c;

    cin >> a >> b >> c;

    if (a + b > c) {

        cout << "YES";

    }

    else {

        cout << "NO";

    }

    return 0;

}

**Зарахування:**



**Pull Request:**

**Висновок:**   
Я навчився використовувати базові алгоритми ітерації по масивами в С++.