

Міністерство освіти і науки України
Національний університет «Львівська політехніка»
Кафедра систем штучного інтелекту



Звіт

про виконання розрахунково-графічних робіт блоку № 7

з дисципліни: «Основи програмування»

до:

Практичних Робіт до блоку № 7

Виконав:

Студент групи ІІІ-11

Яровой Павло Олегович

Мета роботи: одержати практичні навички в розробці і дослідженні алгоритмів розв'язання задач.

1) Умови завдань:

Завдання №1. VNS Practice Work Task 1

Варіант 6. $P = \sqrt{x^2 + b} - b^2 \sin^3(x + a) / xy$, де $b=15,5$; $x=-2,9$;
 $v = \cos^2 x^3 - \frac{x}{\sqrt{a^2 + b^2}}$; $a=1,5$.

Завдання №2. VNS Practice Work Task 2

Варіант 20. Увести координати десяти точок на площині і визначити, в якій чверті більше всього точок.

Завдання №3. VNS Practice Work Task 3

Варіант 11. Обчислення вартості поїздки на автомобілі на дачу (туди і назад). Початковими даними є: відстань до дачі (км); кількість бензину, яка споживає автомобіль на 100 км пробігу; ціна одного літра бензину. Нижче приведений вид екрану під час виконання програми, що рекомендується (дані, які вводяться користувачем, виділені напівжирним шрифтом).

Обчислення вартості поїздки на дачу і назад.

Введіть початкові дані:

Відстань до дачі (км) > **67**

Витрата бензину (літрів на 100 км пробігу) > **8.5**

Ціна літра бензину (грн.) > **6.5**

Поїздка на дачу і назад обійдеться в 74.04 грн.

Реалізувати обчислення струму, по відомих значеннях напруги і опору електричного ланцюга.

Завдання №4. VNS Practice Work Task 4

Варіант 15. Скласти програму, яка обчислює середнє арифметичне послідовності дробових чисел, що вводяться з клавіатури. Після введення останнього числа програма повинна вивести мінімальне і максимальне число послідовності. Кількість чисел послідовності повинна задаватися під час роботи програми.

Завдання №5. Найбільша зростаюча послідовність

Вам задано послідовність із n цілих чисел a_i . Ваша задача — знайти довжину найбільшої зростаючої підпослідовності заданої послідовності.

Вхідні дані

У першому рядку задано ціле число n .

У наступному рядку задано n цілих чисел — послідовність a_i .

Вихідні дані

Виведіть довжину найбільшої зростаючої підпослідовності.

Завдання №6. Верховна рада

Вже не перший рік у нашому суспільстві гостро стоїть питання кількості народних депутатів у Верховній Раді. Утримувати дуже багато дармоїдів-депутатів народ не хоче, проте все має бути справедливо і кожна політична партія повинна отримати кількість місць у парламенті пропорційну до кількості голосів, що вона отримала на виборах. Більш формально, відношення кількості голосів до кількості місць у Верховній Раді має бути однаковим для усіх партій.

В останніх виборах до Верховної Ради взяли участь n партій. Вам відома кількість голосів, що отримала кожна з них. Допоможіть народові мінімізувати загальну кількість депутатських місць.

Вхідні дані

У першому рядку задано одне натуральне число n — кількість партій.

У другому рядку задано n натуральних чисел a_i — кількість голосів, що отримала i -та партія на останніх виборах.

Вихідні дані

У єдиному рядку виведіть одне ціле число — мінімальну загальну кількість депутатських місць у парламенті.

Завдання №7. Цікава гра

Мале Бісеня та Дракон полюбляють проводити дозвілля разом. Сьогодні вони грають в одну дуже цікаву гру.

У них є дошка, що складається з n рядків та m стовпців, всі клітинки якої білі.

Гравці по черзі вибирають одну білу клітинку та зафарбовують її в чорний колір. Бісеня ходить першим. Гравець, який не може зробити хід, тобто на початку ходу якого вся дошка чорна, програє.

Погостривши зубки, Бісеня зрозуміло, що у Дракона велика перевага, адже він двоголовий, а, як то кажуть, «одна голова добре, а дві — краще». Тому воно просить вас допомогти. Вам потрібно сказати за заданими n та m , хто виграє у цій напруженій грі.

Вхідні дані

У єдиному рядку задані два цілих числа n та m — розміри дошки.

Вихідні дані

Єдине слово — **Imp**, якщо переможе Бісеня, та **Dragon**, якщо переможе Дракон.

Завдання №8. Народна вакцина

Зеник дуже любить пити чай, адже це хороша профілактика від реп'яховірусу.

Зеник дуже відповідально ставиться до чаювання, а тому завжди дотримується строгих правил: він п'є чай кожної години, починаючи о 9-тій ранку та закінчуючи о 9-тій вечора, об'єм випитого чаю — завжди ціле число. До того ж, Зеник свято вірить, що для кращого ефекту всі різниці між об'ємами випитого чаю під час двох послідовних чаювань мають бути однаковими протягом усього дня. Тобто об'єми чаю, випитого протягом дня, мають утворювати арифметичну прогресію.

Щоранку Зеник прокидається та планує чаювання на цей день. Для цього він обирає два цілих числа a — об'єм чаю, який потрібно випити о 9-тій годині ранку, та b — об'єм чаю, який потрібно випити о 9-тій вечора.

Далі Зеник рахує, скільки ж чаю він вип'є сумарно за день, якщо дотримуватиметься всіх своїх правил? Ця щоденна процедура виснажує Зеника.

Чи зможете ви написати програму, яка допоможе йому в обчисленнях?

Вхідні дані

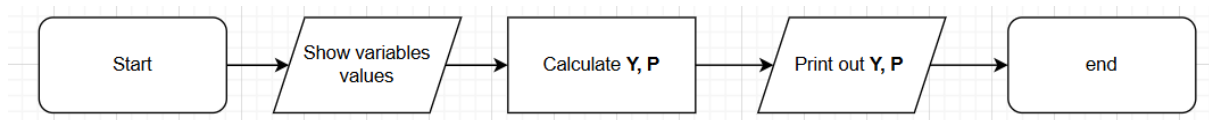
В одному рядку задано два цілих числа a та b — об'єми чаю, які треба випити о 9-ій годині ранку та вечора відповідно.

Вихідні дані

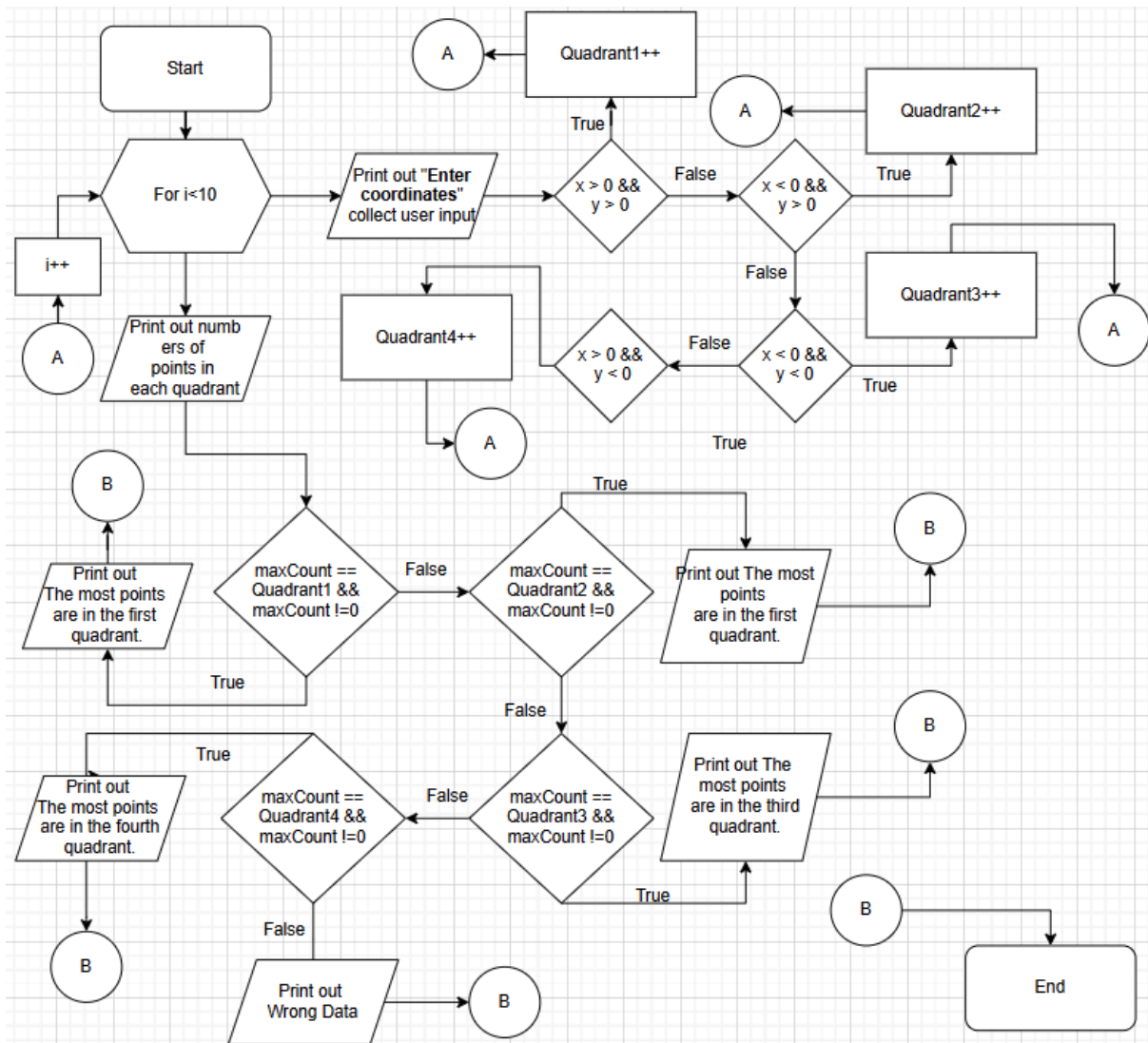
Якщо цілочисельної послідовності, що задовольняє всі вимоги не існує, то виведіть **-1**. Інакше виведіть ціле число — сумарний об'єм випитого за день чаю.

1. Дизайн виконання завдань

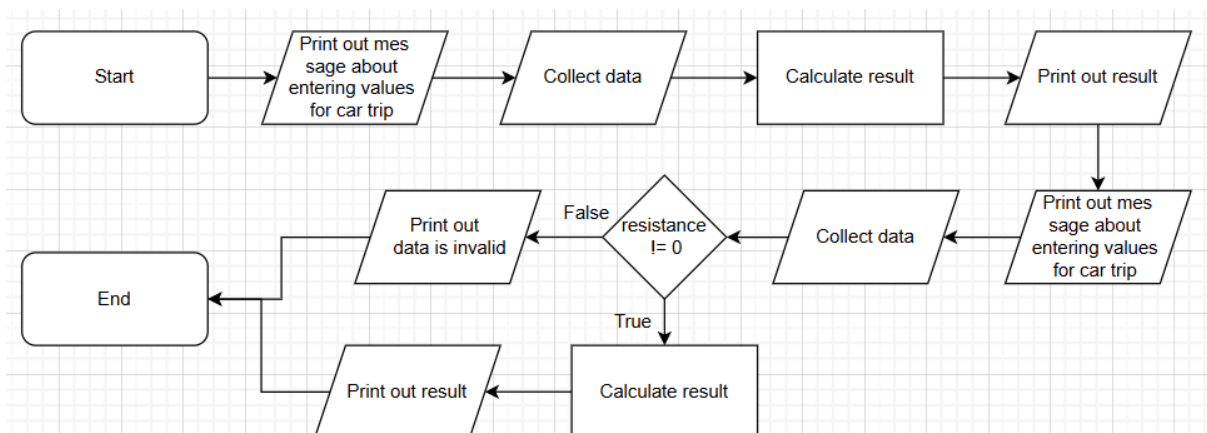
Завдання №1. VNS Practice Work Task 1



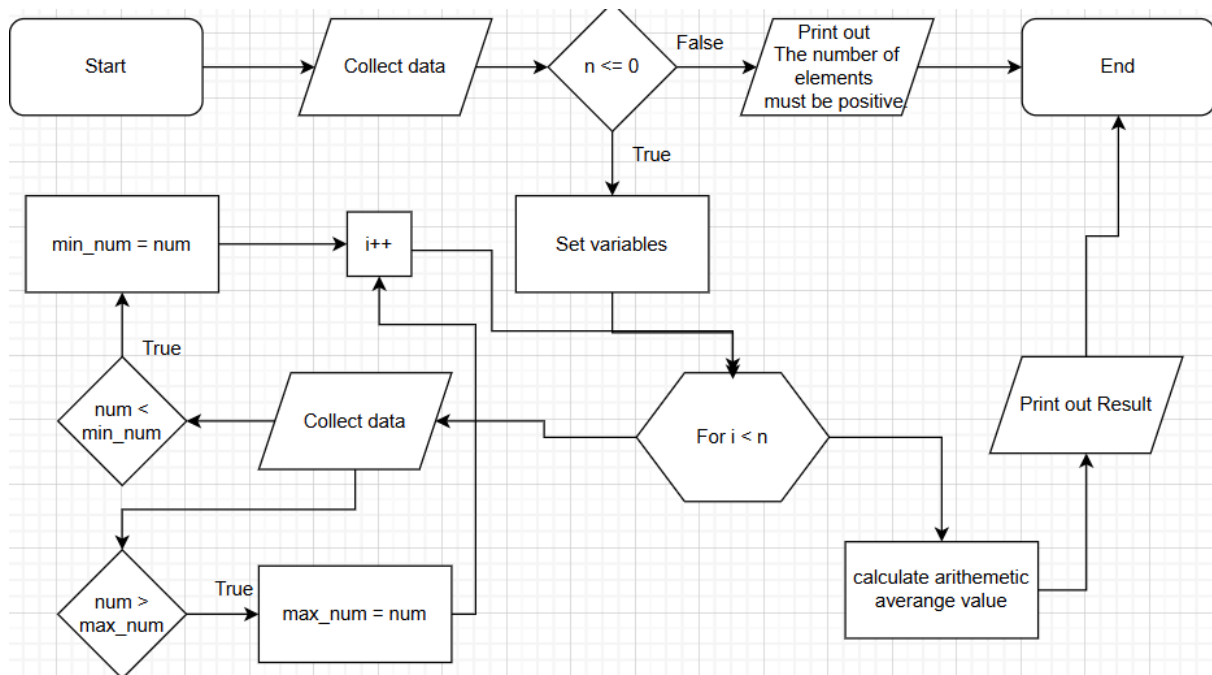
Завдання №2. VNS Practice Work Task 2



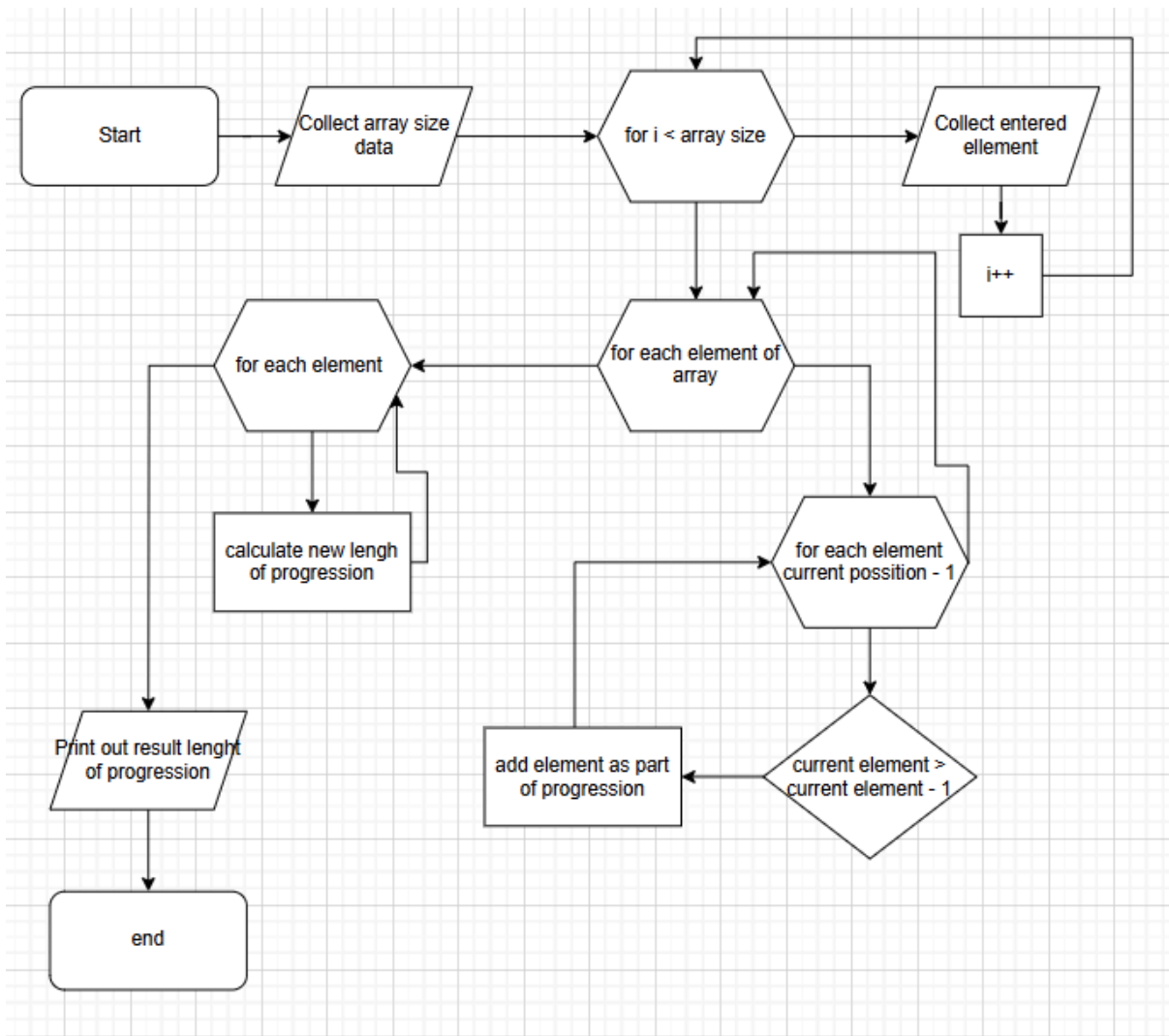
Завдання №3. VNS Practice Work Task 3



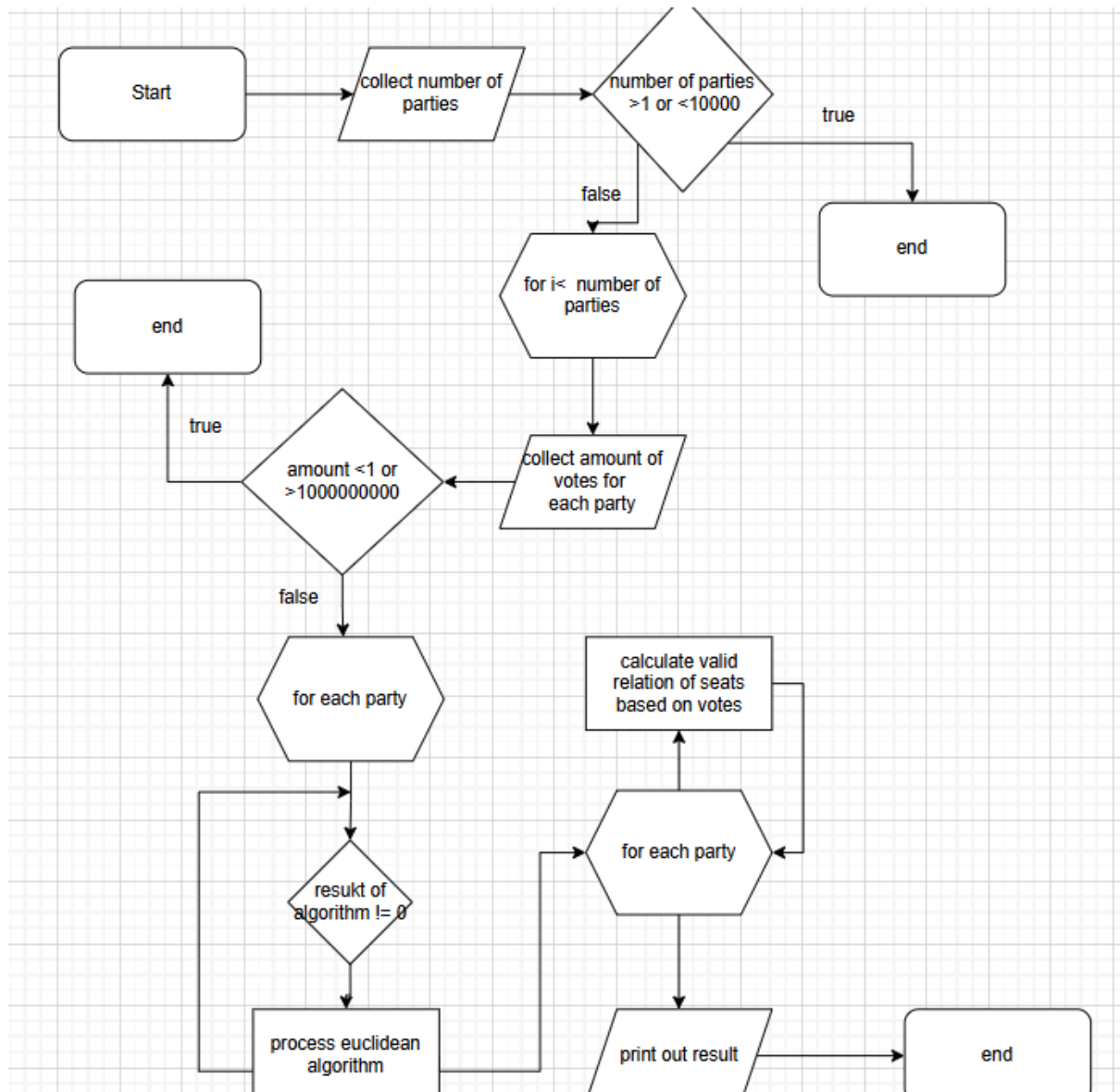
Завдання №4. VNS Practice Work Task 4



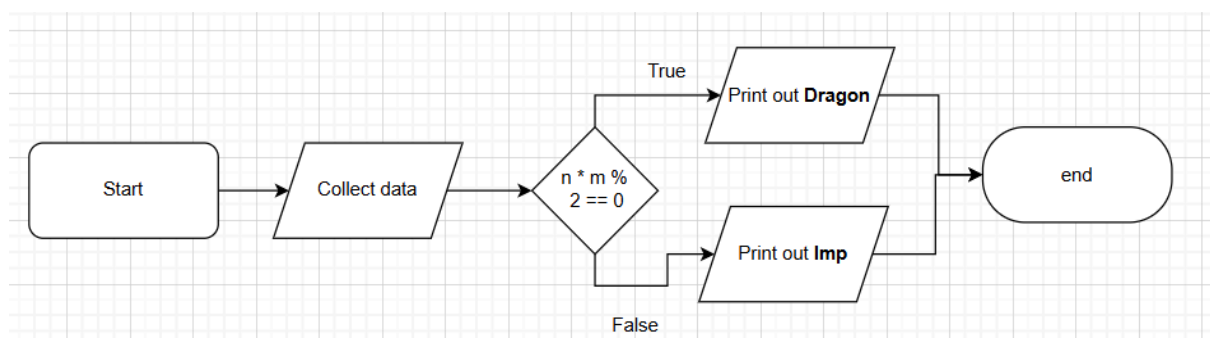
Завдання №5. Найбільша зростаюча послідовність



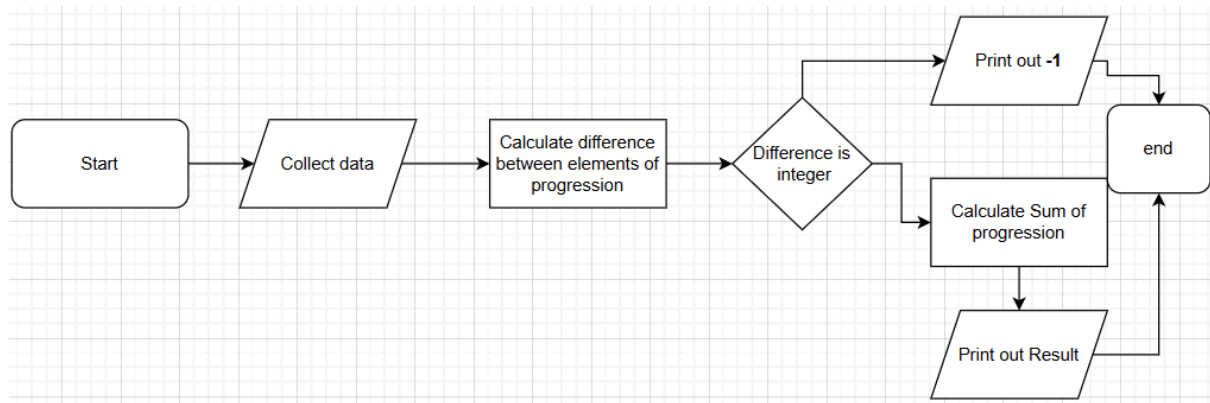
Завдання №6. Верховна рада



Завдання №7. Цікава гра



Завдання №8. Народна вакцина



3 Код програм

Завдання №1. VNS Practice Work Task 1

```

1  #include <iostream>
2  #include <cmath>
3  #include <iomanip>
4
5  using namespace std;
6
7  int main() {
8      double a = 1.5, b = 15.5, x = -2.9;
9
10     cout << "Given values:" << endl;
11     cout << "a = " << a << endl;
12     cout << "b = " << b << endl;
13     cout << "x = " << x << endl;
14
15     double y = pow(cos(pow(x, 3)), 2) - (x / sqrt(pow(a, 2) + pow(b, 2)));
16     double P = sqrt(pow(x, 2) + b) - pow(b, 2) * pow(sin(x + a), 3) / (x * y);
17
18     cout << "Results:" << endl;
19     cout << "y = " << fixed << setprecision(3) << y << endl;
20     cout << "P = " << fixed << setprecision(3) << P << endl;
21
22     return 0;
23 }

```

Завдання №2. VNS Practice Work Task 2

```

1  #include <iostream>
2  #include <algorithm>
3  using namespace std;
4
5  int main() {
6      int Quadrant1 = 0, Quadrant2 = 0, Quadrant3 = 0, Quadrant4 = 0;
7      for (int i = 0; i < 10; i++) {
8          double x, y;
9          cout << "Enter coordinates of point " << i + 1 << " (x, y): ";
10         cin >> x >> y;
11
12         if (x > 0 && y > 0) {
13             Quadrant1++;
14         } else if (x < 0 && y > 0) {
15             Quadrant2++;
16         } else if (x < 0 && y < 0) {
17             Quadrant3++;
18         } else if (x > 0 && y < 0) {
19             Quadrant4++;
20         }
21     }
22     cout << "Number of points in the first quadrant: " << Quadrant1 << endl;
23     cout << "Number of points in the second quadrant: " << Quadrant2 << endl;
24     cout << "Number of points in the third quadrant: " << Quadrant3 << endl;
25     cout << "Number of points in the fourth quadrant: " << Quadrant4 << endl;
26
27     int maxCount = max({Quadrant1, Quadrant2, Quadrant3, Quadrant4});
28
29     if (maxCount == Quadrant1 && maxCount != 0) {
30         cout << "The most points are in the first quadrant." << endl;
31     }
32     else if (maxCount == Quadrant2 && maxCount != 0) {
33         cout << "The most points are in the second quadrant." << endl;
34     }
35     else if (maxCount == Quadrant3 && maxCount != 0) {
36         cout << "The most points are in the third quadrant." << endl;
37     }
38     else if (maxCount == Quadrant4 && maxCount != 0) {
39         cout << "The most points are in the fourth quadrant." << endl;
40     }
41     else {
42         cout << "Wrong data" << endl;
43     }
44     return 0;
45 }

```

Завдання №3. VNS Practice Work Task 3

```

1  #include <iostream>
2  #include <iomanip>
3  using namespace std;
4
5  int main() {
6      double distance, fuelConsumption, fuelPrice;
7      cout << "Calculation of the trip cost to the dacha and back.\n";
8      cout << "Distance to the dacha (km) > ";
9      cin >> distance;
10     cout << "Fuel consumption (liters per 100 km) > ";
11     cin >> fuelConsumption;
12     cout << "Price per liter of fuel (UAH) > ";
13     cin >> fuelPrice;
14
15     double totalCost = (distance * fuelConsumption / 100) * 2 * fuelPrice;
16
17     cout << fixed << setprecision(2);
18     cout << "The trip to the dacha and back will cost " << totalCost << " UAH." << endl;
19
20     double voltage, resistance;
21     cout << "Calculation of current using known values of voltage and resistance.\n";
22     cout << "Enter the voltage (volts) > ";
23     cin >> voltage;
24     cout << "Enter the resistance (ohms) > ";
25     cin >> resistance;
26
27     if (resistance != 0) {
28         double current = voltage / resistance;
29         cout << "The current in the circuit: " << current << " amperes.\n";
30     } else {
31         cout << "Resistance cannot be zero.\n";
32     }
33
34     return 0;
35 }

```

Завдання №4. VNS Practice Work Task 4

```

#include <iostream>
#include <limits>
using namespace std;

int main() {
    int n;
    cin >> n;

    if (n <= 0) {
        cout << "The number of elements must be positive." << endl;
        return 1;
    }

    double sum = 0;
    double min_num = numeric_limits<double>::max();
    double max_num = numeric_limits<double>::min();

    for (int i = 0; i < n; i++) {
        double num;
        cin >> num;

        sum += num;

        if (num < min_num) {
            min_num = num;
        }

        if (num > max_num) {
            max_num = num;
        }
    }

    double average = sum / n;

    cout << "Arithmetic average mean: " << average << endl;
    cout << "Minimum number: " << min_num << endl;
    cout << "Maximum number: " << max_num << endl;

    return 0;
}

```

Завдання №5. Найбільша зростаюча послідовність

```
#include <iostream>
#include <algorithm>
using namespace std;

int main() {
    int n;
    cin >> n;

    int a[n], dp[n];

    for (int i = 0; i < n; i++) {
        cin >> a[i];
        dp[i] = 1;
    }

    for (int i = 1; i < n; i++) {
        for (int j = 0; j < i; j++) {
            if (a[i] > a[j]) {
                dp[i] = max(dp[i], dp[j] + 1);
            }
        }
    }

    int result = 0;
    for (int i = 0; i < n; i++) {
        result = max(result, dp[i]);
    }

    cout << result << endl;

    return 0;
}
```

Завдання №6. Верховна рада

```

#include <iostream>

using namespace std;

int main() {
    int n;
    cin >> n;

    if (n < 1 || n > 1000) {
        return 1;
    }

    int votes[n];

    for (int i = 0; i < n; i++) {
        cin >> votes[i];

        if (votes[i] < 1 || votes[i] > 1000000000) {
            return 1;
        }
    }

    int gcd_val = votes[0];

    for (int i = 1; i < n; i++) {
        int a = gcd_val, b = votes[i];
        while (b != 0) {
            a %= b;
            swap(a, b);
        }
        gcd_val = a;
    }

    long long total_seats = 0;

    for (int i = 0; i < n; i++) {
        total_seats += static_cast<long long>(votes[i]) / gcd_val;
    }

    cout << total_seats << endl;

    return 0;
}

```

Завдання №7. Цікава гра

```
#include <iostream>
using namespace std;

int main() {
    int n, m;
    cin >> n >> m;
    if (n * m % 2 == 0) cout << "Dragon";
    else cout << "Imp";
    return 0;
}
```

Завдання №8. Народна вакцина

```
#include "iostream"
using namespace std;
int main(){
    long long int s;
    long double a1, a13, d;
    cin >> a1 >> a13;

    d = (a13-a1) / 12;
    if (abs(d-int(d)) > 0){
        cout << "-1";
    }
    else{
        s = (13*(a1+a13))/2;
        cout << s;
    }

    return 0;
}
```

4 Результати виконання завдань

Завдання №1. VNS Practice Work Task 1

```
Given values
a = 1.5
b = 15.5
x = -2.9
Results:
y = 0.728
P = -104.037
```

Завдання №2. VNS Practice Work Task 2

```
Enter coordinates of point 1 (x, y): 4 5
Enter coordinates of point 2 (x, y): 8 4
Enter coordinates of point 3 (x, y): 7 1
Enter coordinates of point 4 (x, y): 9 2
Enter coordinates of point 5 (x, y): 11 2
Enter coordinates of point 6 (x, y): -9 7
Enter coordinates of point 7 (x, y): 7 -9
Enter coordinates of point 8 (x, y): -2 -2
Enter coordinates of point 9 (x, y): -7 -6
Enter coordinates of point 10 (x, y): 5 -11
Number of points in the first quadrant: 5
Number of points in the second quadrant: 1
Number of points in the third quadrant: 2
Number of points in the fourth quadrant: 2
The most points are in the first quadrant.
```

Завдання №3. VNS Practice Work Task 3

```
Calculation of the trip cost to the dacha and back.
Distance to the dacha (km) > 200
Fuel consumption (liters per 100 km) > 45
Price per liter of fuel (UAH) > 51
The trip to the dacha and back will cost 9180.00 UAH.
Calculation of current using known values of voltage and resistance.
Enter the voltage (volts) > 220
Enter the resistance (ohms) > 10
The current in the circuit: 22.00 amperes.
```

Завдання №4. VNS Practice Work Task 4

```
2.5
4.1
Arithmetic average mean: 2.3
Minimum number: 0.5
Maximum number: 4.1
```

Завдання №5. Найбільша зростаюча послідовність


```
5
4 7 2 10 3
3
```

3 дні тому	C++ 23	Зараховано	0.003	1.238	Перегляд
------------	--------	------------	-------	-------	--------------------------

Завдання №6. Верховна рада

```
4
3108 1624 5600 2268
450
```

3 дні тому	C++ 23	Зараховано	0.003	1.195	Перегляд
------------	--------	------------	-------	-------	--------------------------

Завдання №7. Цікава гра

```
7 4
Dragon
```

3 дні тому	C++ 23	Зараховано	0.002	0.969	Перегляд
------------	--------	------------	-------	-------	--------------------------

Завдання №8. Народна вакцина

```
1 13
91
```

3 дні тому	C++ 23	Зараховано	0.002	1.207	Перегляд
------------	--------	------------	-------	-------	--------------------------

Висновок: Під час виконання розрахункової роботи я закріпив свої знання у мові C/C++, набуті протягом виконання попередніх завдань.

