Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра систем штучного інтелекту



**Звіт**

про виконання

**Лабораторних та практичних робіт № 2**

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

***з розділу***: «Лінійні та розгалужені алгоритми. Умовні оператори. Константи, змінні»

***Виконав:***

студент групи ШІ-14

Вульчин Володимир Сергійович

# **Тема роботи:**

# Оптимізація програмних процесів: аналіз, порівняння та вдосконалення лінійних та розгалужених алгоритмів з використанням умовних операторів, констант і змінних. Вивчення впливу цих елементів на продуктивність та ефективність програмного забезпечення.

# **Мета роботи:**

1. Ознайомитись з лінійними та розгалуженими алгортимами.
2. Ознайомитись з умовними операторами.
3. Ознайомитись константами та змінними.
4. Виконати завдання: VNS Lab 1 (Завдання 1)
5. Виконати завдання: VNS Lab 1 (Завдання 2)
6. Виконати завдання: VNS Lab 2
7. Виконати завдання: Algotester Lab 1
8. Виконати завдання: Algotester Lab 2
9. Виконати завдання задане на практичних
10. Виконати самостійне завдання по написання коду.

# **Теоретичні відомості:**

1. Теоретичні відомості з переліком важливих тем:

* Тема №1: Лаб# Програмування: VNS Lab 1 (Завдання 1)
* Тема №2: Лаб# Програмування: VNS Lab 1 (Завдання 2)
* Тема №3: Лаб# Програмування: VNS Lab 2 (Завдання 1)
* Тема №4: Лаб# Програмування: Algotester Lab 1 (Завдання 1)
* Тема №5: Лаб# Програмування: Algotester Lab 2 (Завдання 3)
* Тема №6: Практчина# Програмування: Class Practice Task
* Тема №7: Практчина# Програмування: Self Practice Task

1. Індивідуальний план опрацювання теорії:

* Тема №1: Програмування: VNS Lab 1 (Завдання 1)
  + Джерела Інформації:
    - Стаття: [документація до ВНС Лаб 1](https://drive.google.com/drive/u/6/folders/1rywFoh9PJBgf_vJn5EwSyo0Ywo7BvtI3)
    - Команда
  + Що опрацьовано:

Успішно закодено програму, яка за умовою: , при а = 1000 та b = 0.0001 (а i b можуть бути довільними числами), вираховує значення виразу.

* + Статус: Ознайомлений/ Ознайомлений частково / Не ознайомлений
  + Початок опрацювання теми: 16.10.2023
  + Звершення опрацювання теми: 17.10.2023
* Тема №2: Лаб# Програмування: VNS Lab 1 (Завдання 2)
  + Джерела Інформації:
    - Команда
    - Стаття: [документація до ВНС Лаб 1](https://drive.google.com/drive/u/6/folders/1rywFoh9PJBgf_vJn5EwSyo0Ywo7BvtI3)
  + Що опрацьовано:
    - Успішно закодено програму, яка за умовами:

1. m-++n
2. ++m>--n
3. –n<++m
   * Статус: Ознайомлений/ Ознайомлений частково / Не ознайомлений
   * Початок опрацювання теми: 18.10.2023
   * Звершення опрацювання теми: 20.10.2023

* Тема №3: Лаб# Програмування: VNS Lab 2 (Завдання 1)
  + Джерела Інформації:
    - Cтаття: [документація до ВНС Лаб 2](https://drive.google.com/drive/u/6/folders/1rywFoh9PJBgf_vJn5EwSyo0Ywo7BvtI3)
    - Команда
    - Одногрупники
  + Що опрацьовано:
    - Успішно закодено програму, яка за умовою: , при введені користувачем значення n, виводить n результатів, та сумму ряду лише тих результатів n, які менше за .

* + Статус: Ознайомлений/ Ознайомлений частково / Не ознайомлений
  + Початок опрацювання теми: 17.10.2023
  + Звершення опрацювання теми: 23.10.2023
* Тема №4: Лаб# Програмування: Algotester Lab 1 (Завдання 1)
  + Джерела Інформації:
    - Команда.
    - Одногрупники
    - [Відео від індуса](https://www.youtube.com/watch?v=VA8t4ACItsg) (класний чувак, трішки було не зрозуміло, але потім дойшло)
  + Що опрацьовано:

Успішно закодено програму, яка за умовою:

Персонаж 3 рази використовує закляття, кожне з яких може використати хiтпойнти та ману одночасно (які присвоює користувач). У випадку, якщо якесь закляття забирає i хiтпойнти i ману – персонаж мусить програти. Також, якщо в кiнцi персонаж буде мати додатню кiлькiсть хiтпойнтiв та мани – вiн виграє, в iншому випадку програє.

Статус: Ознайомлений/ Ознайомлений частково / Не ознайомлений

* + Початок опрацювання теми: 23.10.2023
  + Звершення опрацювання теми: 29.10.2023
* Тема №5: Лаб# Програмування: Algotester Lab 2 (Завдання 3)
  + Джерела Інформації:
    - Команда.
    - Одногрупники
    - ChatGPT, Bard
  + Що опрацьовано:
    - Успішно закодено програму, яка за умовою:

1. Перевірка на зіткнення до взльоту:

Якщо a1 (позиція лівого дрона) дорівнює aN (позиція правого дрона, де N - розмір масиву), вивести "Collision".

2. Перевірка на "Miss" до взльоту:

Якщо a1 більше за aN, вивести "Miss".

3. Якщо ніяка з вищезазначених умов не виконується, це означає, що дрони зупиняються один навпроти одного, тобто в клітинках ai та ai+1.

У такому випадку вивести "Stopped".

, та виводить у першому рядку фінальну поцизію першого та другого дронів. У другому одне з слів: Collision, Miss, Stopped.

* + Статус: Ознайомлений/ Ознайомлений частково / Не ознайомлений
  + Початок опрацювання теми: 24.10.2023
  + Звершення опрацювання теми: 2.11.2023
* Тема №6: Практчина# Програмування: Class Practice Task
  + Джерела Інформації:
    - Одногрупники
    - Практичне заняття
    - Інформація про оператори: [switch](https://acode.com.ua/urok-68-operator-switch/), [if](http://programming.in.ua/programming/c-plus-plus/281-operators-if-and-switch-c-plus-plus.html), [else if](http://programming.in.ua/programming/c-plus-plus/281-operators-if-and-switch-c-plus-plus.html), [else](http://programming.in.ua/programming/c-plus-plus/281-operators-if-and-switch-c-plus-plus.html). (Вважаю ці сайти одними з найзручніших для вивчення тих, чи інших тем, до того ж пояснено українською мовою)
  + Що опрацьовано:

Реалізувати програму через оператори switch case, if else, if, else, щоб програма змогла вивести при вказаній користувачем погоди, які речі йому слід одягнути.

Статус: Ознайомлений/ Ознайомлений частково / Не ознайомлений

* + Початок опрацювання теми: 18.10.2023
  + Звершення опрацювання теми: 18.10.2023
* №7: Практчина# Програмування: Self Practice Task
  + Що опрацьовано: Реалізовано програму, яка за умови:

Щодня в спортзал приходить певна кількість людей. Необхідно запропонувати користувачеві ввести такі дані: скільки людей відвідувало спортзал в день, ввести вік кожного відвідувача. Та вивести вік найстаршого і наймолодшого з них, а також розрахувати середній вік відвідувачів.

Використано оператори [if](http://programming.in.ua/programming/c-plus-plus/281-operators-if-and-switch-c-plus-plus.html), та цикл [for](https://acode.com.ua/urok-72-tsykl-for/).

* + Статус: Ознайомлений/ Ознайомлений частково / Не ознайомлений
  + Початок опрацювання теми: 19.10.2023
  + Звершення опрацювання теми: 20.10.2023

# **Виконання роботи:**

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

* Лаб# Програмування: VNS Lab 1 (Завдання 1)
* Варіант завдання: 13
* Деталі завдання: Закодити програму, яка за умовою: , при а = 1000 та b = 0.0001, вираховує значення виразу.
* Завдання №2 Лаб# Програмування: VNS Lab 1 (Завдання 2)

Варіант завдання: 13

* Деталі завдання:
  + - Закодити програму, яка буде вирішувати задачу за умов:

1. m-++n (Без особливих труднощів)
2. ++m>--n (Використання операторів if та else if)
3. –n<++m (Використання операторів if та else if)

Де, m та n числові значення, вводяться безпосередньо в консолі.

* Завдання №3 Лаб# Програмування: VNS Lab 2 (Завдання 1)

Варіант завдання: 1

* Деталі завдання:
  + - Успішно закодено програму, яка за умовою: , при введені користувачем значення n, виводить n результатів, та сумму ряду лише тих результатів n, які менше за .

До прикладу: Користувач вводить значення n = 4. Програма виводить обчислені значення, які вміщає в собі n = 4, тобто (n = 1, n = 2, n = 3 до n = 4).

Наступним кроком, програма виводить «Суму рядку меншу за », якщо таких значень не існує, то виводиться число «0»

* Завдання №4 Лаб# Програмування: Algotester Lab 2 (Завдання 3)

Варіант завдання: 3

* Деталі завдання:
* Завдання №4 Практчина# Програмування: Class Practice Task

Варіант завдання:

* Деталі завдання:

Реалізувати програму через оператори switch case, if else, if, else, щоб програма змогла вивести при вказаній користувачем погоди, які речі йому слід одягнути та чим зайнятись.

* Завдання №6 Практчина# Програмування: Self Practice Task

Варіант завдання (відсутній)

* Деталі завдання:

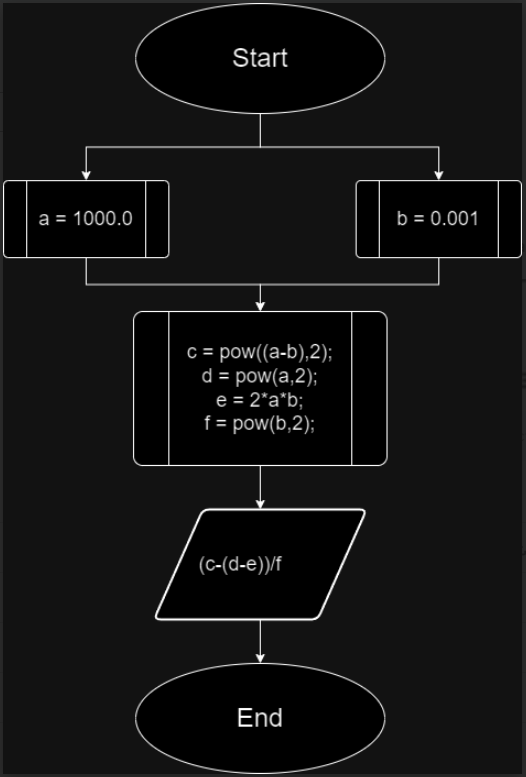
Щодня в спортзал приходить певна кількість людей. Необхідно запропонувати користувачеві ввести такі дані: скільки людей відвідувало спортзал в день, ввести вік кожного відвідувача. Та вивести вік найстаршого і наймолодшого з них, а також розрахувати середній вік відвідувачів.

Використано оператори [if](http://programming.in.ua/programming/c-plus-plus/281-operators-if-and-switch-c-plus-plus.html), та цикл [for](https://acode.com.ua/urok-72-tsykl-for/).

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

Програма №1 VNS Lab 1 (Завдання 1)

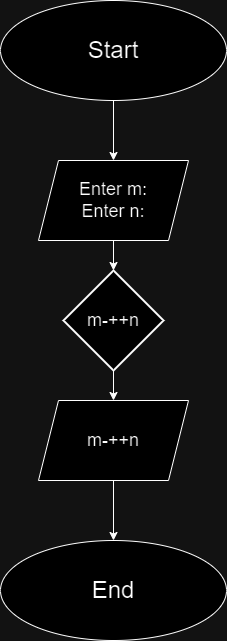
Блок-схема:



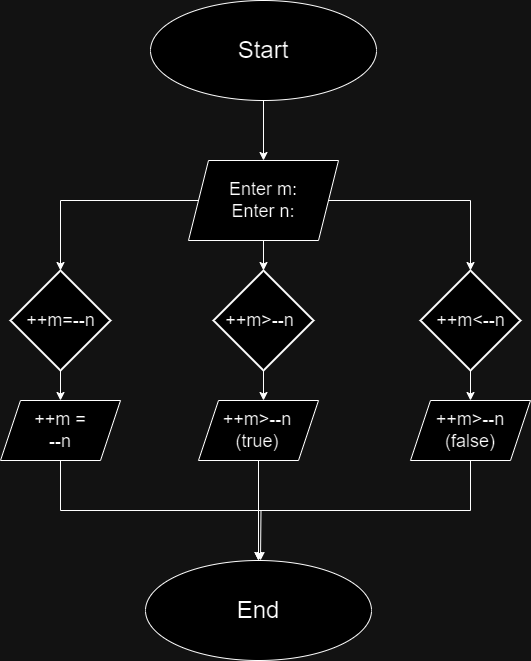
* Планований час на реалізацію: 35 хвилин.

Програма №2 Лаб# Програмування: VNS Lab 1 (Завдання 2)

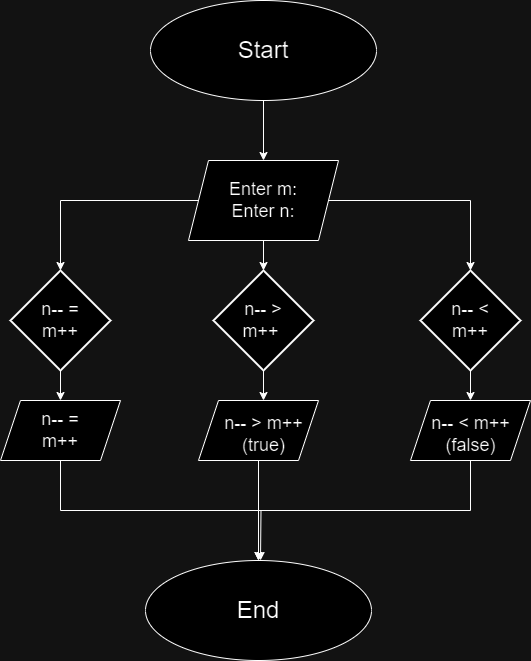
* Блок-схема (1):



* Блок-схема (2):



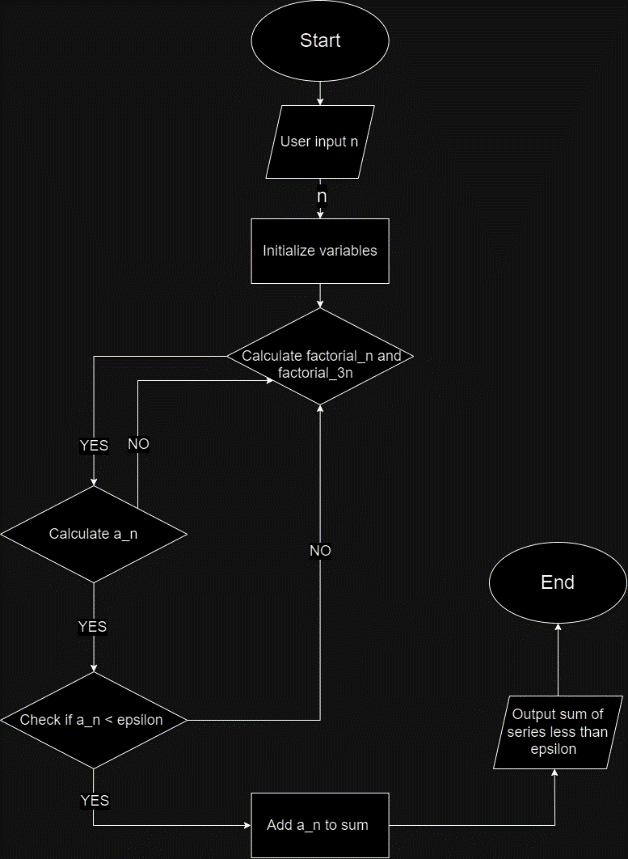
* Блок-схема (3):



* Планований час на реалізацію: 1.5 години.

Програма №3: Лаб# Програмування: VNS Lab 2 (Завдання 1)

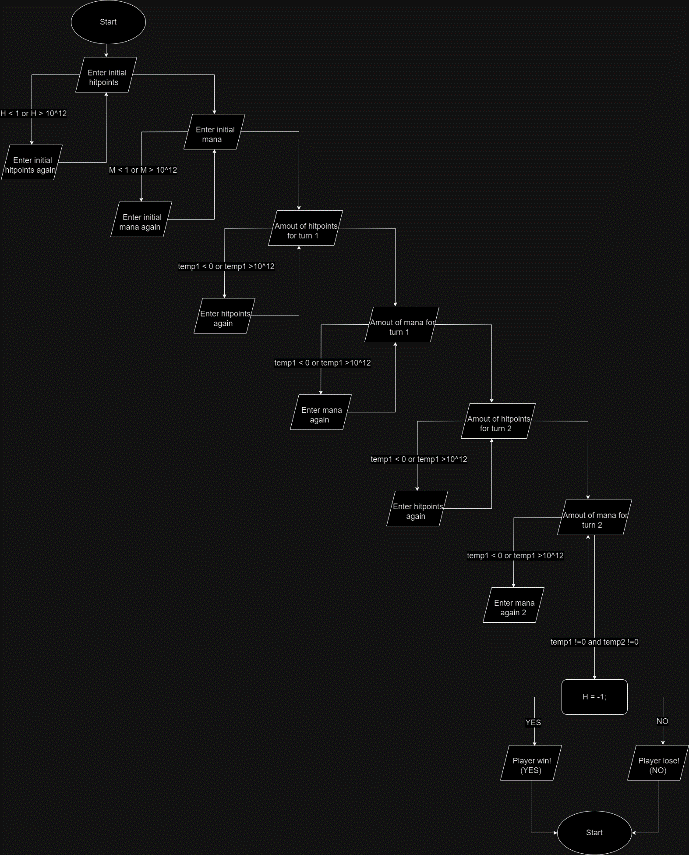
* Блок-схема



* Планований час на реалізацію ~12 годин.

Програма №4: Лаб# Програмування: Algotester Lab 1 (Завдання 1)

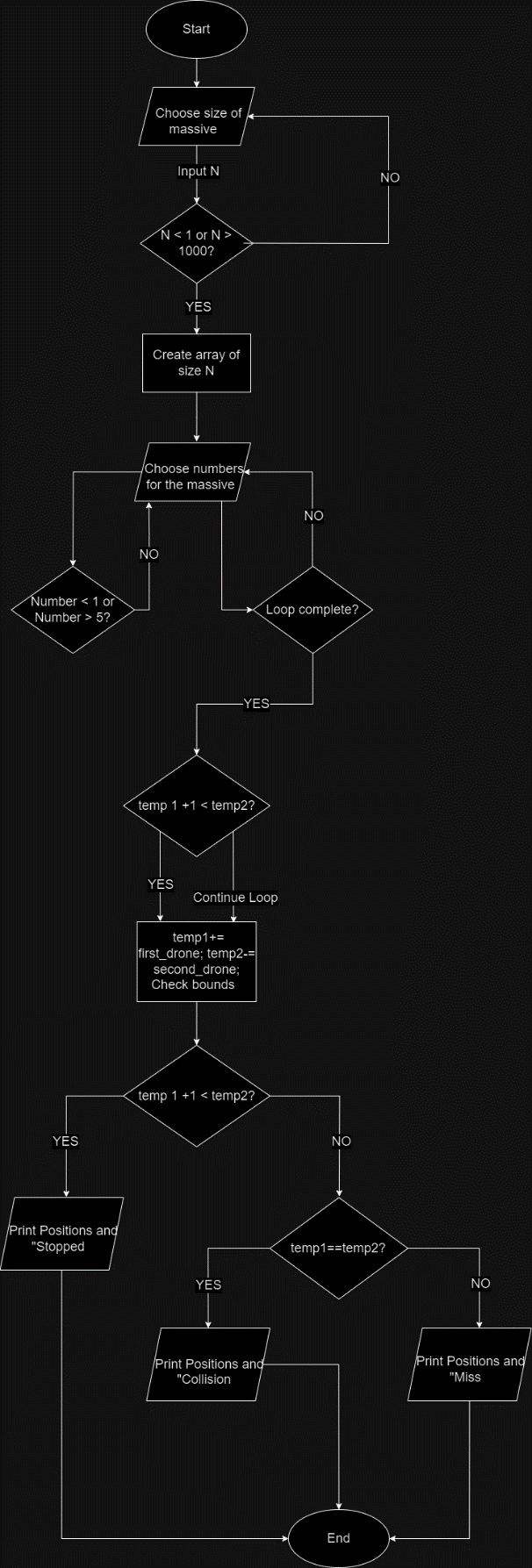
* Блок-схема



* Планований час на реалізацію ~16 годин.

Програма №5: Лаб# Програмування: Algotester Lab 2 (Завдання 3)

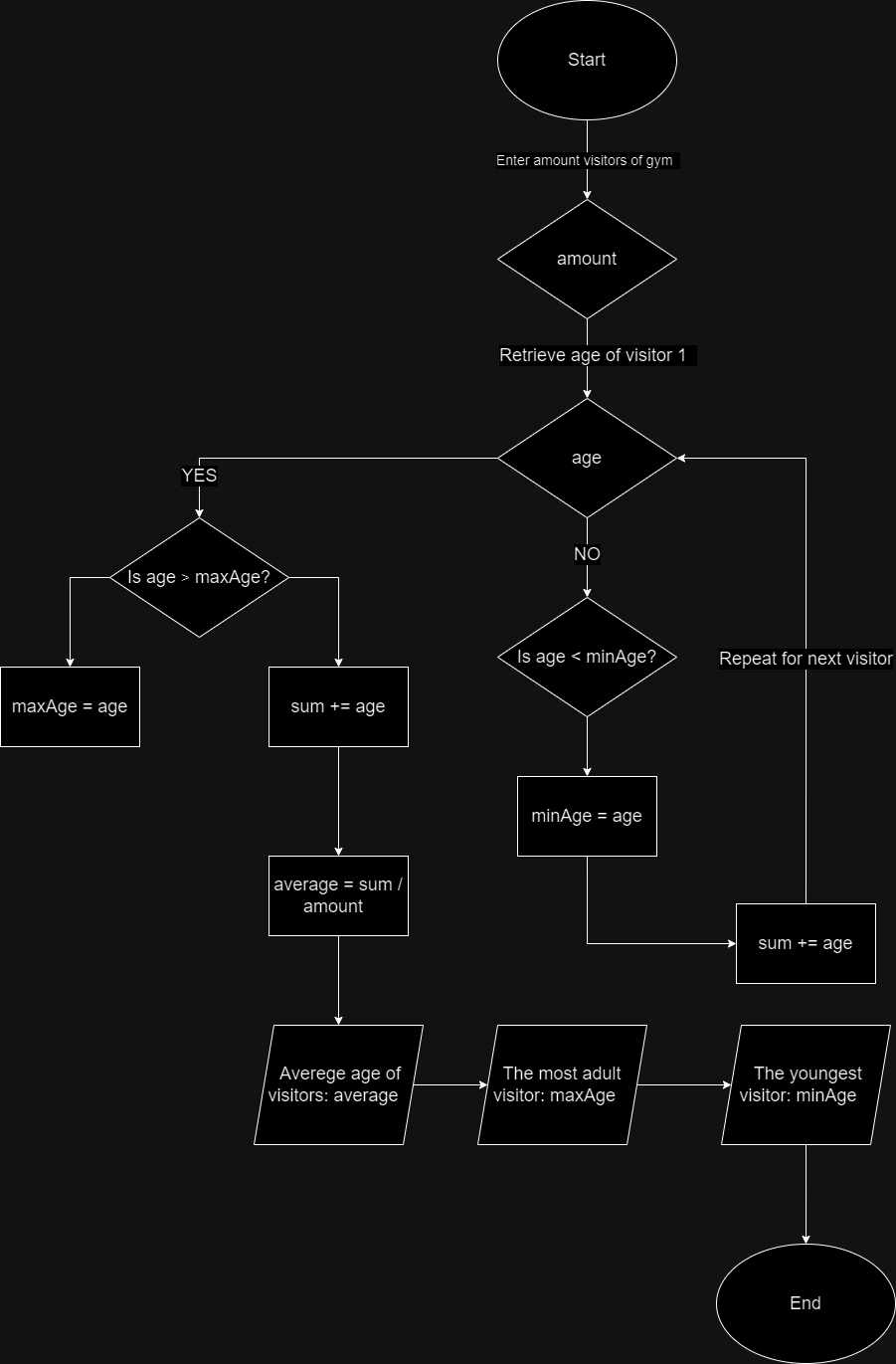
* Блок-схема



* Планований час на реалізацію 20 годин.

Програма №7: Практчина# Програмування: Self Practice Task

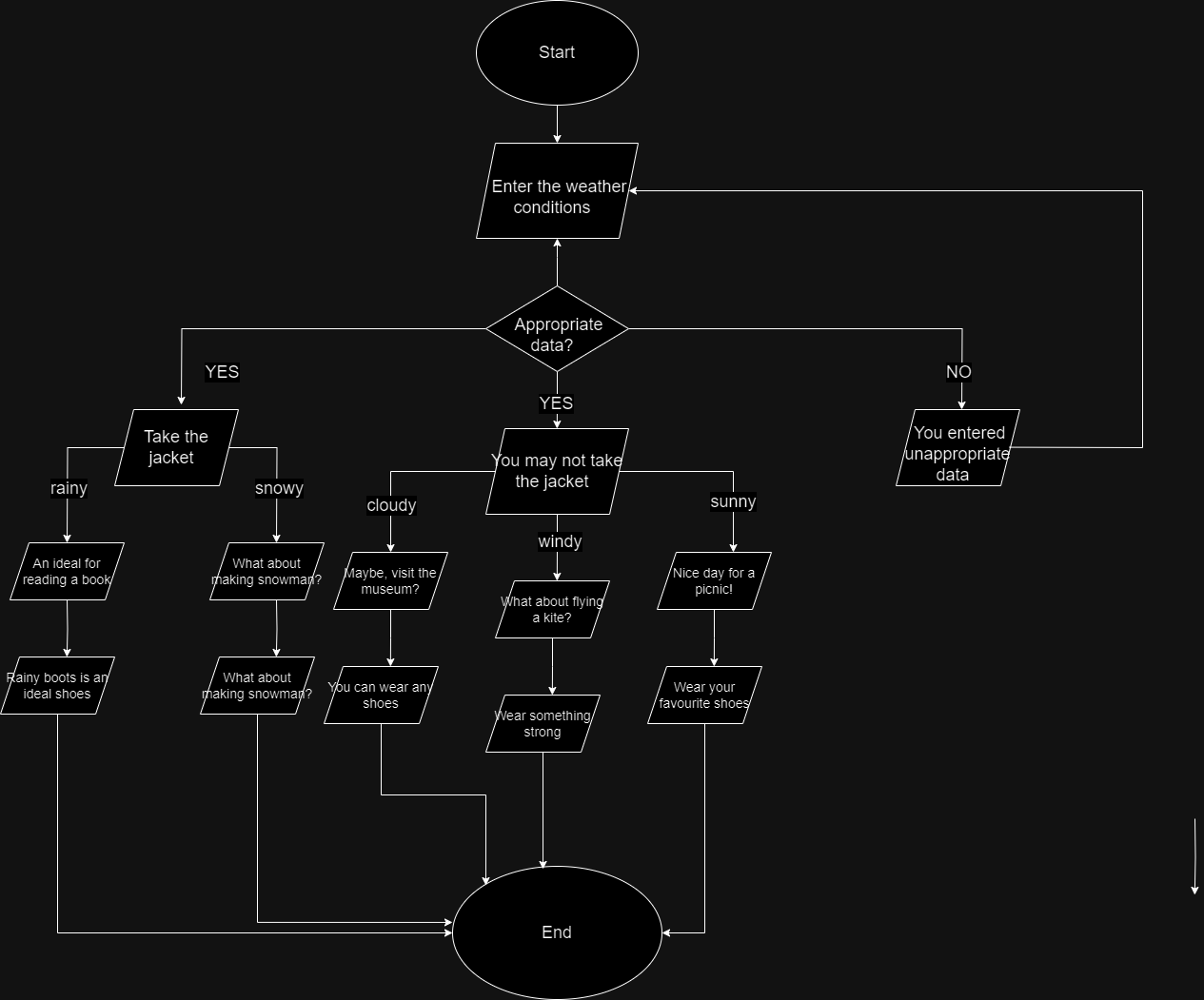
* Блок-схема



* Планований час на реалізацію 3,5 години.

Програма №6: Практчина# Програмування: Class Practice Task

* Блок-схема



* Планований час на реалізацію 4,5 години.

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

Завдання №1: Лаб# Програмування: VNS Lab 1 (Завдання 1)

Посилання на файл програми у пул-запиті GitHub: https://github.com/artificial-intelligence-department/ai\_programming\_playground/pull/238

#include <iostream>

#include <cmath>

    using namespace std;

        int main(){

            long double a, b, c, d, e, f;

            a = 1000.0;

            b = 0.0001;

            // Виражаю покрокові дії через змінні c, d, e, f.

            // Рядок коду нижче, є індентичним виразу: (a-b)^2(a^2-2ab)/b^2

            c = pow((a-b),2);

            d = pow(a,2);

            e = 2\*a\*b;

            f = pow(b,2);

            cout << "Відповідь: " << (c-(d-e))/f;

            return 0;

        }

Завдання №2: Лаб# Програмування: VNS Lab 1 (Завдання 2)

Посилання на файл програми у пул-запиті GitHub: https://github.com/artificial-intelligence-department/ai\_programming\_playground/pull/238

// 1) m-++n

#include <iostream>

using namespace std;

int main(){

        int m, n;

        cout << "Enter m: " << endl;

        cin >> m;

        cout << "Enter n: " << endl;

        cin >> n;

//  m-++n = означає що ми від числа M віднімає число N збільшене на одиницю

        cout << "Знайчення виразу дорівнює: " << (m-(++n));

return 0;

}

// 2) ++m>--n

#include <iostream>

using namespace std;

int main(){

        int m, n;

        cout << "Enter m: " << endl;

        cin >> m;

        cout << "Enter n: " << endl;

        cin >> n;

        ++m;

        --n;

//  ++m>--n = означає що значення m збільшене на 1, має бути більшим за значення n зменшене на 1

        // Використовую оператор if, з умовою (m > n), якщо рівність справджується, то виводить cout нижче.

        if (m > n){

        cout << "Знайчення виразу m > за значення виразу n. " << (m > n) << endl;

        // Використовую оператор else if, з умовою (m < n), за умови , якщо умова вище несправджується, і виводить cout нижче.

        } else if (m < n){

        cout << "Знайчення виразу m < за значення виразу n. " << (m < n) << endl;

        // Використовую вдруге оператор else if, з умовою (m = n), за умови , якщо обидві умови вище несправджуються, і виводить cout нижче.

        } else if (m == n){

        cout << "Знайчення виразу m = значення виразу n. " << (m = n) << endl;

        }

return 0;

}

// 3) n-- > m++

#include <iostream>

using namespace std;

int main(){

    int m, n;

    cout << "Enter m: " << endl;

    cin >> m;

    cout << "Enter n: " << endl;

    cin >> n;

    --n;

    ++m;

//  n-- > m++ = означає що значення n зменшене на 1, має бути більшим за значення m збільшне на 1.

// Використовую оператор if, з умовою (n > m), якщо рівність справджується, то виводить cout нижче.

if (n > m) {

    cout << "The value of n > for the value of m " << (n > m) << endl;

// Використовую оператор else if, з умовою (n < m), за умови , якщо умова вище несправджується, і виводить cout нижче.

} else if (n < m) {

    cout << "Value of n < expression per value of m expression " << (n < m) << endl;

// Використовую вдруге оператор else if, з умовою (n = m), за умови , якщо обидві умови вище несправджуються, і виводить cout нижче.

} else if (n = m) {

    cout << "The value of n = the value of m. " << (n == m) << endl;

}

return 0;

}

Завдання №3: Лаб# Програмування: VNS Lab 2 (Завдання 1)

Посилання на файл програми у пул-запиті GitHub: https://github.com/artificial-intelligence-department/ai\_programming\_playground/pull/238

#include <iostream>

#include <math.h>

using namespace std;

// n! = n\*(n-1)!

// a\_n = (3^n\*n!)/(3\*n)!

// a\_n+1 / an = 1/9n^2+9n+2

int main()

{

    int temp; // тимчасове число, яке = n

    double epsilon = 0.0001; // значення епсілом

    double sum = 0; // сума всіх чисел

    double a\_n; // відповідь при якомусь n, якщо n = 1, то відповідь: 0.5, якщо n = 2, то відповідь: 0.025. Тобто, якщо введешь n = 2, то воно виведе n = 1 і n = 2.

    double factorial\_n = 1; // виведення n!

    double factorial\_3n = 1; // виведення (3\*n)!

    int n; // значення, яке вводить користувач

    cout << "Введіть значення n: ";

    cin >> n;

    for(int i = 1 ; i <= n; i++){ // цикл, який відбувається n разів, int i = 1 - починаємо з числа 1, при умові i <= n - цикл продовжується до тих пір поки i > n; i++ - виконується в самому кінці

    factorial\_n = 1; // вказали значення 1, щоб забути факторіал попереднього n.

    factorial\_3n = 1; // вказали значення 1, щоб забути факторіал попереднього n.

    temp = i; // робиму копію i, якщо шукаємо n = 1 то temp = 1, і т.д.

    for (int i = 1; i <= temp; i++){ // цикл, який відбувається temp разів, int i = 1 - починаємо з числа 1, при умові i <= temp - цикл продовжується до тих пір поки i > temp; i++ - виконується в самому кінці

        factorial\_n \*= i; // домножуємо до факторіалу i temp разів.

    }

    for (int i = 1; i <= temp \* 3; i++){ // цикл, який відбувається temp разів, int i = 1 - починаємо з числа 1, при умові i <= temp \* 3 - цикл продовжується до тих пір поки i > temp \* 3; i++ - виконується в самому кінці

        factorial\_3n \*= i; // домножуємо до факторіалу i (temp \* 3) разів

    }

    a\_n = (factorial\_n \* pow(3,temp)) / factorial\_3n; // підрахунок нашої формули на даних момент a\_n при 1 = 0.5 і т.д.

    cout << a\_n << " "; // виводимо а\_n

    if(a\_n < epsilon) { // перевіряємо чи а\_n є меншим за епсілон

    sum += a\_n; // якщо а\_n є меншим за епсілон, то ми додаємо його до суми, якщо  не а\_n є меншим за епсілон то ми його пропускаємо.

    }

    }

    cout << "\nСума ряду менша за epsilon: " << sum;

    return 0;

}

Завдання №4: Лаб# Програмування: Algotester Lab 1 (Завдання 1)

Посилання на файл програми у пул-запиті GitHub: https://github.com/artificial-intelligence-department/ai\_programming\_playground/pull/238

#include <iostream>

#include <string>

#include <cmath>

using namespace std;

int main()

{

    int H, M, temp1, temp2, hp = 0, mn = 0; // H, M - початкове хп, temp1, temp2 - затрати хп і мани на ході, hp = 0, mn = 0 - загальні витрати мани.

    cout << "Enter your initial hitpoints: ";

    cin >> H;

    while(H < 1 || H > pow(10,12)){ // умова виконання

        cout << "Enter your initial hitpoints again: ";

        cin >> H;

    }

    cout << "Enter your initial mana: ";

    cin >> M;

    while(M < 1 || M > pow(10,12)){ // умова виконання

        cout << "Enter your mana initial again: ";

        cin >> M;

    }

    cout << "Enter amount of hitpoints you will spend on turn 1: ";

    cin >> temp1;

    while(temp1 < 0 || temp1 > pow(10,12)){ // умова виконання

        cout << "Enter your hitpoints again: ";

        cin >> temp1;

    }

    hp += temp1; // додавання загального хп і хп, яке було вказане під час ходу 1

    cout << "Enter amount of mana you will spend on turn 1: ";

    cin >> temp2;

    while(temp2 < 0 || temp2 > pow(10,12)){ // умова виконання

        cout << "Enter your mana again: ";

        cin >> temp2;

    }

    mn += temp2; // додавання загальної мани і мант, яка була вказана під час ходу 1

    if(temp1 != 0 && temp2 != 0){ // умова виконання

        H =-1; // Якщо, хоча б одне з них не дорівнює 0, то максимальне значення хп стає -1

    }

    cout << "Enter amount of hitpoints you will spend on turn 2: ";

    cin >> temp1;

    while(temp1 < 0 || temp1 > pow(10,12)){ // умова виконання

        cout << "Enter your hitpoints again: ";

        cin >> temp1;

    }

    hp += temp1; // додавання загального хп і хп, яке було вказане під час ходу 2

    cout << "Enter amount of mana you will spend on turn 2: ";

    cin >> temp2;

    while(temp2 < 0 || temp2 > pow(10,12)){ // умова виконання

        cout << "Enter your mana again: ";

        cin >> temp2;

    }

    mn += temp2; // додавання загальної мани і мант, яка була вказана під час ходу 2

    if(temp1 != 0 && temp2 != 0){

        H = -1; // Якщо, хоча б одне з них не дорівнює 0, то максимальне значення хп стає -1

    }

    cout << "Enter amount of hitpoints you will spend on turn 3: ";

    cin >> temp1;

    while(temp1 < 0 || temp1 > pow(10,12)){ // умова виконання

        cout << "Enter your hitpoints again: ";

        cin >> temp1;

    }

    hp += temp1; // додавання загального хп і хп, яке було вказане під час ходу 3

    cout << "Enter amount of mana you will spend on turn 3: ";

    cin >> temp2;

    while(temp2 < 0 || temp2 > pow(10,12)){ // умова виконання

        cout << "Enter your mana again: ";

        cin >> temp2;

    }

    mn += temp2; // додавання загальної мани і мант, яка була вказана під час ходу 3

    if(temp1 != 0 && temp2 != 0){ // умова виконання

        H = -1; // Якщо, хоча б одне з них не дорівнює 0, то максимальне значення хп стає -1

    }

    if((H - hp) >= 0 && (M - mn) >= 0){ // умова виконання

        cout << "YES (Player win!)";

    }else{

        cout << "NO (Player lose!)";

    }

}

Завдання Тема №5: Лаб# Програмування: Algotester Lab 2 (Завдання 3)

Посилання на файл програми у пул-запиті GitHub: https://github.com/artificial-intelligence-department/ai\_programming\_playground/pull/238

#include <iostream>

#include <string>

using namespace std;

int main()

{

    int N, first\_drone = 0, second\_drone = 0, temp1 = 0, temp2; // N - розмір масиву, temp1 - тимчасова позиція дрона, пишем 0 бо числа додаються, temp2 - є знизу, 1 ми пишем коли є множення чисел

    cout << "Choose size of massive: ";

    cin >> N;

    while(N < 1 || N > 1000){ // обмеження масиву від 1 до 1000

        cout << "Choose size of massive again (from 1 to 1000): ";

        cin >> N;

    }

    int\* array = new int[N]; // значення масиву, пишем \* бо ми вводим елемент масиву

    cout << "Choose numbers for the massive: " << endl;

    for (int i = 0; i < N; i++) { // починаючи з 0 елемента, доки елемент менший за масив, перейти на кожен наступний елемент

        cout << "Choose " << i+1 << " number: ";

        cin >> array[i]; // ввід елемента масиву

        while(array[i] < 1 || array[i] > 5){ // елемент масиву має бути від 1 до 5

            cout << "Choose " << i+1 << " number again (from 1 to 5): ";

            cin >> array[i];

        }

    }

    temp2= N-1; // оскільки елемент масиву починається з 0

    for (int i = 0; i < N; i++) {

        if (temp1+1 < temp2) { // коли вони поряд

            temp1+= first\_drone; // додавання позицій, де розташований перший дрон

            temp2-= second\_drone;  // додавання позицій, де розташований другий дрон

            if(temp1 > N-1){ // перший дрон вилетів за масив зправа

                temp1=N-1;

            }

            if(temp2 < 0){ // другий дрон вилетів за масив зліва

                temp2=0;

            }

            first\_drone = array[temp1]; // елемент на якому зупиниться перший дрон

            second\_drone = array[temp2]; // елемент на якому зупиниться другий дрон

        } else {

            break;

        }

    }

    if(temp1+1==temp2){ // дрони один біля одного

        cout << "Position of first\_drone: " << temp1+1 << endl << "Position of second\_drone: " << temp2+1<<endl;

        cout << "Stopped";

    }else{

        if(temp1==temp2){ // дрони врізались

        cout << "Position of first\_drone: " << temp1+1 << endl << "Position of second\_drone: " << temp2+1<<endl;

        cout << "Collision";

        }else{ // один з дронів/дрони перелетів

        cout << "Position of first\_drone: " << temp1+1 << endl << "Position of second\_drone: " << temp2+1<<endl;

        cout << "Miss";

        }

    }

    return 0;

}

Завдання №6: Практчина# Програмування: Class Practice Task

Посилання на файл програми у пул-запиті GitHub: https://github.com/artificial-intelligence-department/ai\_programming\_playground/pull/238

#include <iostream>

#include <string>

using namespace std;

int main()

{

    string weather;

    cout << "Enter the weather conditions: ";

    getline(cin, weather); // використвуємо функцію getline, адже без неї, буде зчитуватись лише перше слово/буква після першого пробілу.

    if (weather != "cloudy" && weather != "snowy" && weather != "rainy" && weather != "windy" && weather != "sunny")

    {

        cout << "You entered unappropriate data";

    }

    else

    {

        char i = weather[0];

        if (weather == "snowy" || weather == "rainy")

        {

            cout << "You must take the jacket" << endl;;

        }

        else

        {

            cout << "You have not take the jacket" << endl;;

        }

        if (weather == "sunny")

        {

            cout << "Nice day for a picnic!" << endl;

        }

        else if (weather == "rainy")

        {

            cout << "An ideal for reading a book" << endl;

        }

        else if (weather == "cloudy")

        {

            cout << "Maybe, visit the museum?" << endl;

        }

        else if(weather == "snowy")

        {

            cout << "What about making snowman?" << endl;

        }

        else

        {

            cout << "What about flying a kite?" << endl;

        }

        switch(i)

        {

        case 's':

            if (weather == "sunny")

            {

                cout << "Wear your favourite shoes" << endl;

            }

            else

            {

                cout << "Snow boots will warm up your foots" << endl;

            }

            break;

        case 'r':

            cout << "Rainy boots is an ideal shoes" << endl;

            break;

        case 'c':

            cout << "You can wear any shoes" << endl;

            break;

        case 'w':

            cout << "Wear something strong" << endl;

            break;

}

    }

}

Завдання Тема №7: Практчина# Програмування: Self Practice Task

Посилання на файл програми у пул-запиті GitHub: https://github.com/artificial-intelligence-department/ai\_programming\_playground/pull/238

#include <iostream>

using namespace std;

int main()

{

 int age = 0; // вводить користувач

 int maxAge = 0; // для запису максимального значення віку

 int minAge = 100; // для запису мінімального значення віку

 int sum = 0; // загальна сума, для розрахунків

 int average = 0; // для запису середнього віку

 int amount = 0; // кількість відвідувачів спортзалу

cout << "Enter amount visitors of gym: ";

cin >> amount;

for (int i = 0; i < amount; i++)

{

 cout << "Enter age " << i + 1 << " visitor: "; // запит на введення числа

cin >> age;

if (age > maxAge) // якщо воно більше, ніж зберігає в собі змінна max

maxAge = age; // записуємо в неї це число

if (age < minAge)

minAge = age;

 sum += age; // накопичення загальної сумми

}

 average = sum / amount; // для підрахунку середнього віку

cout << "\nAverege age of visitors: " << average << endl;

cout << "\nThe most adult visitor: " << maxAge << endl;

cout << "\nThe youngest visitor: " << minAge << endl;

return 0;

}

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

Завдання №1: Лаб# Програмування: VNS Lab 1 (Завдання 1)

Output:

Відповідь: 1.00117

Затрачений час: 35 хв.

Завдання №2: Лаб# Програмування: VNS Lab 1 (Завдання 2)

1)

Input:

Enter m:

5

Enter n:

5

Output:

Знайчення виразу дорівнює: -1

2)

Input:

Enter m:

9

Enter n:

9

Output:

Знайчення виразу m > за значення виразу n. 1

Input:

Enter m:

9

Enter n:

11

Output:

Знайчення виразу m = за значення виразу n. 1

Input:

Enter m:

9

Enter n:

12

Output:

Знайчення виразу m < за значення виразу n. 1

3)

Input:

Enter m:

9

Enter n:

9

Output:

Value of n < expression per value of m expression 1

Input:

Enter m:

7

Enter n:

9

Output:

The value of n = the value of m. 1

Input:

Enter m:

8

Enter n:

11

Output:

The value of n > for the value of m 1

Затрачений час: 1.5 години.

Завдання №3: Лаб# Програмування: VNS Lab 2 (Завдання 1)

Input:

Введіть значення n: 5

Output:

0.5 0.025 0.000446429 4.05844e-06 2.22991e-08

Сума ряду менша за epsilon: 4.08074e-06

Затрачений час: 12 годин.

Завдання №4: Лаб# Програмування: Algotester Lab 1 (Завдання 1)

Input:

Enter your initial hitpoints: 100 100

Enter your initial mana: Enter amount of hitpoints you will spend on turn 1: 10

Enter amount of mana you will spend on turn 1: 5

Enter amount of hitpoints you will spend on turn 2: 10

Enter amount of mana you will spend on turn 2: 5

Enter amount of hitpoints you will spend on turn 3: 10

Enter amount of mana you will spend on turn 3: 5

Output:

NO (Player lose!)

Затрачений час: 16 годин.

Завдання №5: Лаб# Програмування: Algotester Lab 2 (Завдання 3)

Input:

Choose size of massive: 5

Choose numbers for the massive:

Choose 1 number: 3

Choose 2 number: 4

Choose 3 number: 1

Choose 4 number: 3

Choose 5 number: 4

Output:

Position of first\_drone: 4

Position of second\_drone: 1

Miss

Затрачений час: 20 годин.

Завдання №6: Практчина# Програмування: Class Practice TaskEnter the weather conditions: sunny

Input:

Enter the weather conditions: sunny

Output:

You have not take the jacket

Nice day for a picnic!

Wear your favourite shoes

Input:

Enter the weather conditions: snowy

Output:

You must take the jacket

What about making snowman?

Snow boots will warm up your foots

Input:

Enter the weather conditions: windy

Output:

You have not take the jacket

What about flying a kite?

Wear something strong

Input:

Enter the weather conditions: rainy

Output:

You must take the jacket

An ideal for reading a book

Rainy boots is an ideal shoes

Input:

Enter the weather conditions: cloudy

Output:

You have not take the jacket

You can wear any shoes

Input:

Enter the weather conditions: hello

Output:

You entered unappropriate data

Затрачений час: 3.5 годин.

Завдання №7: Практчина# Програмування: Self Practice Task

Input:

Enter amount visitors of gym: 5

Enter age 1 visitor: 35

Enter age 2 visitor: 23

Enter age 3 visitor: 17

Enter age 4 visitor: 19

Enter age 5 visitor: 29

Output:

Averege age of visitors: 24

The most adult visitor: 35

The youngest visitor: 17

Затрачений час: 4.5 годин.

# **Висновки:**

У ході проведеного дослідження з оптимізації програмних процесів, було здійснено аналіз, порівняння та вдосконалення лінійних та розгалужених алгоритмів з використанням умовних операторів, констант і змінних. Результати дослідження свідчать про важливий вплив цих елементів на продуктивність та ефективність програмного забезпечення.

Виявлено, що правильне використання умовних операторів, оптимізація констант та ефективне керування змінними можуть значно покращити продуктивність програм. Розгалужені алгоритми, зокрема, можуть бути доречними в складних сценаріях, де необхідно враховувати різні умови і діяти відповідно.

Дослідження також показало, що оптимізація алгоритмів із врахуванням контексту програми і вибір правильних типів змінних може великою мірою впливати на час виконання програм та використання ресурсів.