Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра систем штучного інтелекту

A blue and white logo

Description automatically generated

**Звіт**

**про виконання лабораторних та практичних робіт блоку № 2**

На тему:  « Лінійні алгоритми. Розгалужені алгоритми. Умовні та логічні оператори. Системи числення. Змінні. Константи. Типи даних. Розмір Типів Даних (Двійкова система). Ввід вивід. Базові операції та вбудовані функції. Коментарі.»

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

до:

ВНС Лабораторної Роботи № 1

Алготестер Лабораторної Роботи № 1

Практичних Робіт № 2

**Виконав:**

Студент групи ШІ-13

Кібиш Тарас Юрійович

# **Тема роботи:**

Знайомство з С/С++. Виконання програм простої структури. Використання основних операторів мови С/С++

# **Мета роботи:**

Здобування професійних навичок в роботі з лінійними та розгалужними алгоритмами, умовними операторами, константантами і зміними.

# **Теоретичні відомості:**

1. Теоретичні відомості з переліком важливих тем:

* Тема №1: Управління вимогами (розуміти завдання) та проектування .
* Тема №2: Lab# programming: VNS Lab 1 Task 1.
* Тема №3: Lab# programming: VNS Lab 1 Task 2.
* Тема №4: Lab# programming: Algotester Lab 1 Task 2
* Тема №5: Practice# programming: Class Practice Task
* Тема №6: Practice# programming: Self Practice Task

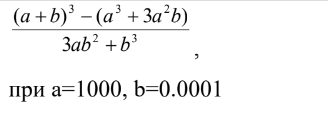
1. Індивідуальний план опрацювання теорії:

* . Тема №1: Управління вимогами (розуміти завдання) та проектування .
  + Джерела Інформації
    - <https://www.visual-paradigm.com/tutorials/flowchart-tutorial/>
    - <https://www.programiz.com/article/flowchart-programming>
  + Що опрацьовано:
    - Створено діаграми для усіх заданих завдань
  + Статус: Ознайомлений
  + Початок опрацювання теми: 14.10.2023
  + Звершення опрацювання теми: 9.11.2023
* Тема №2: Lab# programming: VNS Lab 1 Task 1.
  + Джерела Інформації:
    - [https://chat.openai.com/](https://chat.openai.com/%20)
    - Лекції
  + Що опрацьовано:
    - Створено код для розвязання Vns Lab 1 Task 1
  + Статус: Ознайомлений
  + Початок опрацювання теми: 14.10.2023
  + Звершення опрацювання теми: 26.10.2023
* Тема №3: Lab# programming: VNS Lab 1 Task 2
  + Джерела Інформації:
  + Що опрацьовано:
    - Створено код для розвязання Vns Lab 1 Task 2
  + Статус: Ознайомлений
  + Початок опрацювання теми: 14.10.2023
  + Звершення опрацювання теми: 26.10.2023
* Тема №4: Lab# programming: Algotester Lab 1 Task 2
  + Джерела Інформації:
    - <https://www.programiz.com/cpp-programming/examples/array-largest-element>
    - <https://chat.openai.com/>
  + Що опрацьовано:
    - Створено код для розвязання Algotester Lab 1 Task 2
  + Статус: Ознайомлений
  + Початок опрацювання теми: 14.10.2023
  + Звершення опрацювання теми: 31.10.2023
* Тема №5: Practice# programming: Class Practice Task
  + Джерела Інформації:
  + Що опрацьовано:
    - Створено код для розвязання Class Practise task
  + Статус: Ознайомлений
  + Початок опрацювання теми: 14.10.2023
  + Звершення опрацювання теми: 31.10.2023
* Тема №6: Practice# programming: Self Practice Task
  + Джерела Інформації:
  + Що опрацьовано:
    - Створено код для розвязання Self Practise Task
  + Статус: Ознайомлений
  + Початок опрацювання теми: 14.10.2023
  + Звершення опрацювання теми: 31.10.2023

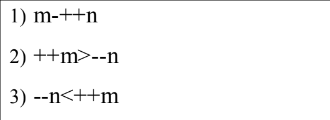
# **Виконання роботи:**

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

Завдання №2 Lab# programming: VNS Lab 1 Task 1

* Варіант завдання: 14
* Деталі завдання :
* 
* Важливі деталі для врахування в імплементації програми:
* Для вводу й виводу даних використати операції >> й << і стандартні потоки
* cin й cout.
* 2. Для обчислення степеня можна використати функцію pow(x,y) з
* бібліотечного файлу math.h.
* 3. При виконанні завдання 1 треба використати допоміжні змінні для зберігання
* проміжних результатів.
* Наприклад: c=pow(a,3);d=3\*a\*a\*b;e=3\*a\*b\*b;f=pow(b,3);

Завдання №3 Lab# programming: VNS Lab 1 Task 2

* Варіант завдання:14
* Деталі завдання :
* 
* Важливі деталі для врахування в імплементації програми
* 1. Для вводу й виводу даних використати операції >> й << і стандартні потоки
* cin й cout.
* Завдання №4 Lab# programming: Algotester Lab 1 Task 2
* Варіант завдання:2
* Деталі завдання :
* Обмеження: 1 сек., 256 МiБ
* У вас є стiл, у якого є 4 нiжки, довжини яких вам дано.
* Ви хочете зробити нiжки рiвної довжини, для цього ви вiдпиляєте d вiд кожної нiжки (тобто
* вам буде дано 4 числа, кожне з яких буде означати вiдпилювання вiд вiдповiної нiжки стола).
* Якщо пiд час вiдпилювання найдовша нiжка стола буде у 2 рази бiльша-рiвна нiж найменша
* нiжка - стiл перевернеться, але вiдпилювати нiжки це вам не завадить.
* Тобто якщо hmax >= 2 ∗ hmin то стiл перевертається. Увага, це може статися i мiж початком
* та кiнцем вiдпилювання, наприклад коли вiдпиляють 2, але ще не встигнуть вiдпиляти 3тю нiжку.
* Також ми вважаємо що перед вiдпилюванням стiл не перевернеться.
* Ваше завдання сказати чи пiсля усiх манiпуляцiй стiл буде цiлий та паралельний пiдлозi.
* Якщо довжина, яку вiдрiжуть буде бiльша за довжину ножки - вам треба вивести ERROR.
* Увага! Навiть якщо стiл перевернеться - ви все одно вiдпилюєте нiжки i можете отримати
* ERROR.
* Важливі деталі для врахування в імплементації програми :
* Обмеження
* 0 ≤ h1,2,3,4 ≤ 10^12
* 0 ≤ d1,2,3,4 ≤ 10^12
* Завдання №5 Practice# programming: Class Practice Task
* Варіант завдання
* Деталі завдання:
* Epic 2 Practice - Оператори галуження
* Ви створюєте простий порадник щодо погоди. Користувач вводить поточні погодні умови, а програма видає рекомендації щодо активності на основі погоди.
* Вимоги
* Запропонувати користувачеві ввести поточні погодні умови.
* Варіанти
* ⁃ sunny;
* ⁃ rainy;
* ⁃ cloudy;
* ⁃ snowy;
* ⁃ windy;
* Якщо користувач вводить будь-яку іншу умову, запропонуйте йому ввести дійсну умову.
* Використовуйте таку логіку
* if else - щоб вирішити, чи повинен користувач взяти куртку чи ні.
* if, else if - щоб надати рекомендацію щодо активності (прогулянка, футбол, настільні ігри, etc).
* switch case - для визначення типу рекомендованого взуття.
* Деталі логіки
* Рішення чи брати куртку (використовуючи if else)
* Якщо йде сніг або дощ, користувач повинен одягнути куртку.
* В іншому випадку куртка не потрібна.
* Рекомендація щодо активності (використання if, else if)
* Якщо сонячно, порекомендуйте «Чудовий день для пікніка!».
* Інакше, якщо буде дощ, рекомендуємо «Ідеальна погода, щоб читати книгу всередині!».
* Інакше, якщо хмарно, рекомендуємо «Може, відвідати музей?».
* Інакше, якщо сніг, порекомендуйте «Як щодо того, щоб зліпити сніговика?».
* Інакше, якщо буде вітер, порекомендуйте «Запустіть повітряного змія, якщо він у вас є!».
* Рекомендації щодо взуття (з використанням футляра для вимикача)
* sunny -> "Взуй улюблені кросівки!"
* rainy -> "Дощові чоботи - гарна ідея!"
* cloudy -> "Сьогодні підходить будь-яке взуття."
* snowy -> "Снігові черевики зігріють ваші ноги!"
* windy -> "Одягніть щось міцне!"
* Важливі деталі для врахування в імплементації програми

Завдання №6: Practice# programming: Self Practice Task   
- Варіант завдання

* Деталі завдання
* Персонажу по одному дають сторони 55 кубів a1..5, з яких він будує піраміду.
* Коли він отримує куб з ребром ai - він його ставить на існуючий, перший ставить на підлогу (вона безмежна).
* Якщо в якийсь момент об’єм куба у руці (який будуть ставити) буде більший ніж у куба на вершині піраміди - персонаж програє і гра закінчується. Розмір усіх наступних кубів після програшу **не враховується**.
* Тобто якщо ai−1<ai - це програш.
* Ваше завдання - сказати як закінчиться гра.

# Input

* 5 цілих чисел a1..a5 - сторони кубів

# Output

* Існуючі варіанти:
* **LOSS** - якщо персонаж не зможе поставити куб.
* **WIN** - якщо персонаж зможе поставити усі куби
* **ERROR** - якщо сторона куба буде мати неможливу довжину, тобто ai <= 0
* Важливі деталі для врахування в імплементації програми
* −1012≤ai≤1012

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

Програма №2 Lab# programming: VNS Lab 1 Task 1.

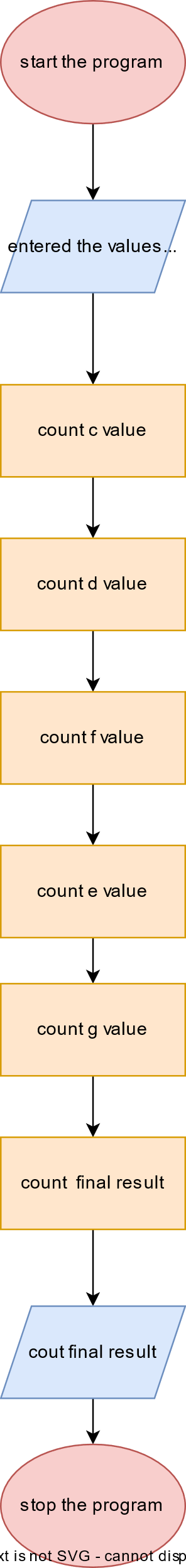
* Блок-схема

Рисунок :№2 Lab# programming: VNS Lab 1 Task 1

* Планований час на реалізацію: 1 година
* Важливі деталі для врахування в імплементації

Програма №3 Lab# programming: VNS Lab 1 Task 2

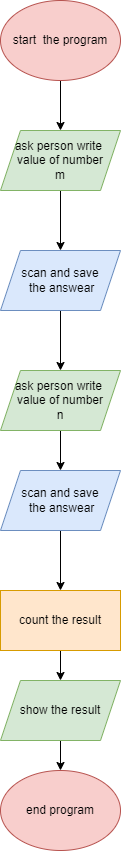
* Блок-схема
* 

Рисунок :№3 Lab# programming: VNS Lab 1 Task 2

* Планований час на реалізацію: 1 година
* Важливі деталі для врахування в імплементації

Програма №4 Lab# programming: Algotester Lab 1 Task 2

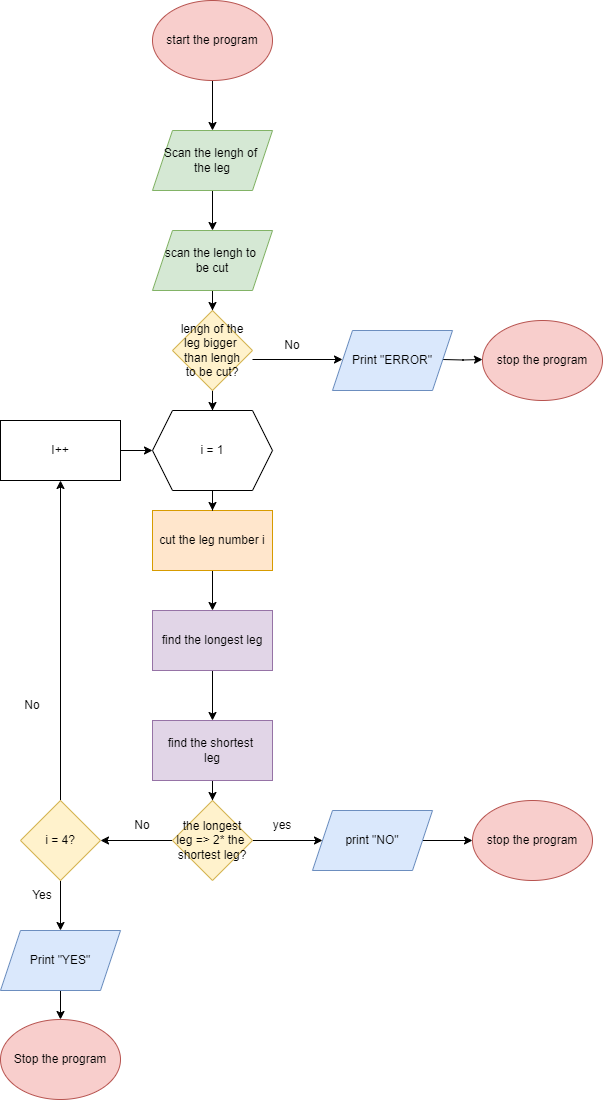
* Блок-схема
* 

Рисунок :№4 Lab# programming: Algotester Lab 1 Task 2

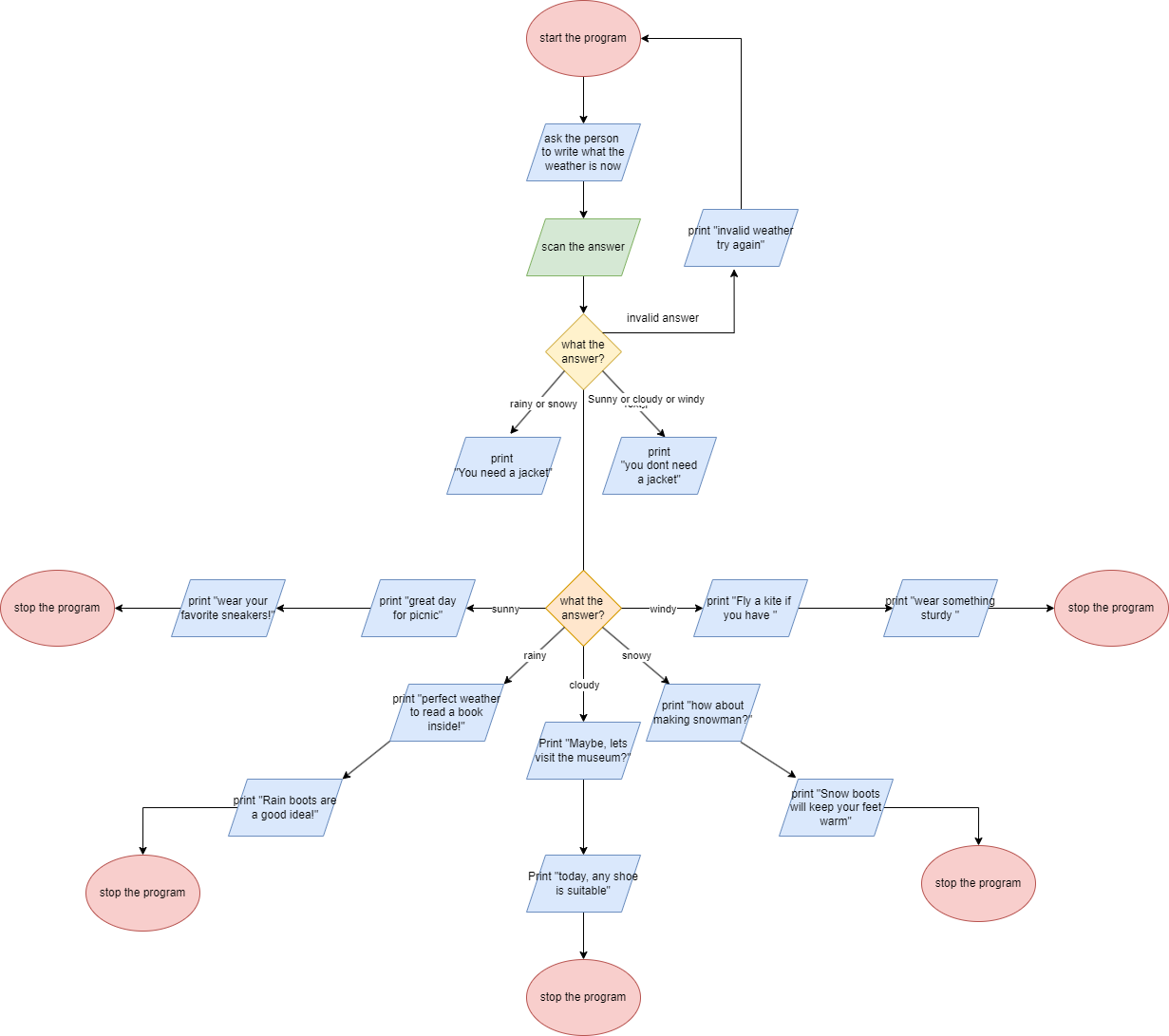
* Планований час на реалізацію 1 день
* Важливі деталі для врахування в імплементації
* Програма №5: Practice# programming: Class Practice Task
* Блок-схема

Рисунок : №5: Practice# programming: Class Practice Task

* Планований час на реалізацію 80 хв

Важливі деталі для врахування в імплементації

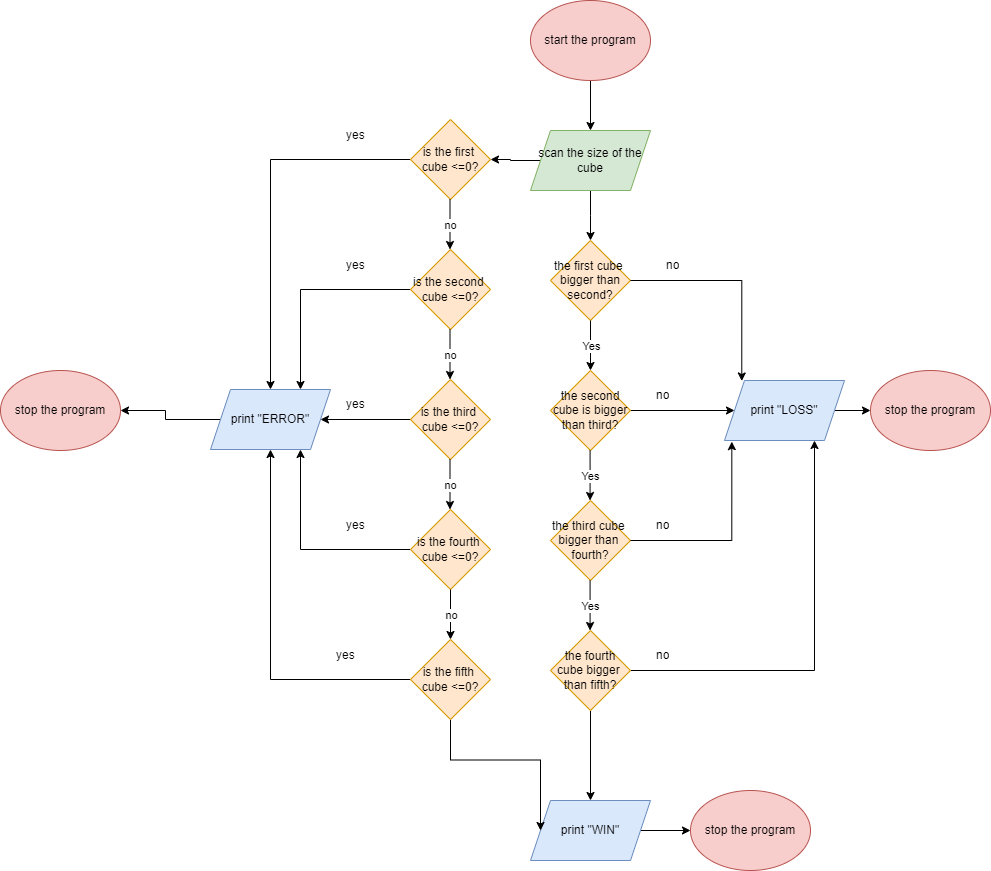
* Програма №6: Practice# programming: Self Practice Task
* Блок-схема
* 

Рисунок :№6: Practice# programming: Self Practice Task

* Планований час на реалізацію 1 день

Важливі деталі для врахування в імплементації

## **3. Конфігурація середовища до виконання завдань:**

Завдання №\_\_ Деталі по конфігурації середовища + скріншоти з підписами до скріншотів.

Підпис та № до блоку з скріншотами до конфігурації

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

Завдання №2: Lab# programming: VNS Lab 1 Task 1

<https://github.com/artificial-intelligence-department/ai_programming_playground/blob/epic_2_practice_and_labs_taras_kibysh/ai_13/taras_kibysh/Epic%202%20-%20Taras%20Kibysh/vns_lab/vns_lab_1_task_1_taras_kibysh.cpp>

#include <cmath>

#include <iostream>

*int* main(){

using *namespace* std;

*float* a = 1000;

*float* b = 0.0001;

*float* c,d,f,e,g;

*float* result;

    c = pow((a + b),3);

    d = pow(a,3);

    f = 3\*pow(a,2)\*b;

    e = 3\*a\*pow(b,2);

    g = pow(b,3);

    result = (c - (d+f))/(e+g);

    cout<<result;

    return 0;

}

Підпис та № до блоку з кодом програми

Завдання №3: Lab# programming: VNS Lab 1 Task 2

<https://github.com/artificial-intelligence-department/ai_programming_playground/blob/epic_2_practice_and_labs_taras_kibysh/ai_13/taras_kibysh/Epic%202%20-%20Taras%20Kibysh/vns_lab/vns_lab_1_task_2_taras_kibysh.cpp>

#include <iostream>

*int* main(){

    using *namespace* std;

*float* m;

*float* n;

cout<<"write number m:";

cin>> m;

cout<<"write number n:";

cin>> n;

*float* result1 =(m+--n);

*float* result2 =(m++<++n);

*float* result3 =(n--< --m);

cout<<result1<<endl;

cout<<result2<<endl;

cout<<result3<<endl;

return 0;

}

Завдання №4: Lab# programming: Algotester Lab 1 Task 2

<https://github.com/artificial-intelligence-department/ai_programming_playground/blob/epic_2_practice_and_labs_taras_kibysh/ai_13/taras_kibysh/Epic%202%20-%20Taras%20Kibysh/Algotester_lab/algotester_lab_1_task_2_taras_kibysh.cpp>

#include <iostream>

#include <cmath>

#include <algorithm>

*int* main(){

    using *namespace* std;

// Вводим змінні

// довжина ніжків

*long* *long* *int* a[4];

//довжина відрізання

*long* *long* *int* l[4];

// скануєм довжину ніжків

for(*int* i = 0;i < 4; i++){

cin >> a[i];

}

// скануєм довжину скільки потрібно відрізати

for(*int* i = 0;i < 4; i++){

cin >> l[i];

}

//виводим Error в разі неможливих даних

if(a[0]< l[0] ||a[1]< l[1] ||a[2]< l[2] ||a[3]< l[3]){

    cout <<"ERROR";

    return 0;

}

// вводим зміну міннімальна довжина

*long* *long* *int* minl = a[0];

//вводим зміну максимальної довжини

*long* *long* *int* maxl = a[0];

for(*int* j = 0; j < 4; j++){

    //відрізаєм кожну ножку по порядку

        a[j] = a[j]-l[j];

    //шукаєм найменшу довжину в масиві

    minl = \*min\_element(a, a+4);

    //шукаєм найбільшу довжину в масиві

    maxl = \*max\_element(a, a+4);

    //порівнюєм довжини

    if(maxl>= minl\*2 || minl == 0){

        cout <<"NO";

        return 0;

}

}

cout <<"YES";

return 0;

}

Завдання №5: Practice# programming: Class Practice Task

<https://github.com/artificial-intelligence-department/ai_programming_playground/blob/epic_2_practice_and_labs_taras_kibysh/ai_13/taras_kibysh/Epic%202%20-%20Taras%20Kibysh/practice/class_practice_2_taras_kibysh.cpp>

#include <iostream>

#include <string>

*int* main(){

using *namespace* std;

string weather;

while(true){

cout << "Write types of weather:";

cin >> weather;

cout  << weather<< endl;

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

cout << "invalid weather try again"<<endl;

}

else

break;

}

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

cout << "You need a jacket\n";

else

cout << "you dont need a jacket\n";

if(weather == "sunny")

cout << "Great day for picnic\n";

else if(weather == "rainy")

cout << "Perfect weather to read a book inside!\n";

else if(weather == "cloudy")

cout << "Maybe, lets visit the museum?\n";

else if(weather == "snowy")

cout << "How about making a snowman?\n";

else if(weather == "windy")

cout << "Fly a kite if you have one!\n";

// яке взуття взути за допомогою switch

*char* symbol = weather[0];

switch(symbol){

 case 'r':

 cout << "Rain boots are a good idea!";

 break;

 case 'c':

 cout << "Today, any shoe is suitable.";

 break;

 case 'w':

 cout << "Wear something sturdy!";

 break;

 case 's':

 if(weather == "sunny")

 cout << "Wear your favorite sneakers!";

 else

 cout << "Snow boots will keep your feet warm!" ;

 break;

}

return 0;

}

Завдання №6: Practice# programming: Self Practice Task

<https://github.com/artificial-intelligence-department/ai_programming_playground/blob/epic_2_practice_and_labs_taras_kibysh/ai_13/taras_kibysh/Epic%202%20-%20Taras%20Kibysh/practice/self_practice_2_taras_kibysh.cpp>

#include <cstdio>

*int* main() {

*long* *long* *int* a1, a2, a3, a4, a5;

    scanf("%lli", &a1);

    scanf("%lli", &a2);

    scanf("%lli", &a3);

    scanf("%lli", &a4);

    scanf("%lli", &a5);

    if (a1 <= 0){

    printf("ERROR");

    return 0;

    }

    else if(a1 < a2){

    printf("LOSS");

    return 0;}

     if (a2 <= 0){

    printf("ERROR");

    return 0;}

    else if(a2 < a3){

    printf("LOSS");

    return 0;

    }

     if (a3 <= 0){

    printf("ERROR");

    return 0;

     }

    else if(a3 < a4){

    printf("LOSS");

    return 0;

    }

     if (a4 <= 0){

    printf("ERROR");

    return 0;

     }

    else if(a4 < a5){

    printf("LOSS");

    return 0;

    }

    else if (a5 <= 0){

    printf("ERROR");

    return 0;

    }

     printf("WIN");

return 0;

}

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

Завдання №2: Lab# programming: VNS Lab 1 Task 1

0.997384

Підпис та № до блоку з виконанням та тестуванням програми

Час затрачений на виконання завдання 30 хв

Завдання №3: Lab# programming: VNS Lab 1 Task 2

write number m:10

write number n:9

18

0

1

Підпис та № до блоку з виконанням та тестуванням програми

Час затрачений на виконання завдання 30 хв

Завдання №4: Lab# programming: Algotester Lab 1 Task 2

10 10 10 10

5 5 5 5

NO



Підпис та № до блоку з виконанням та тестуванням програми

Час затрачений на виконання завдання 7 днів

Завдання №5: Practice# programming: Class Practice Task

Write types of weather:sunny

sunny

you dont need a jacket

Great day for picnic

Wear your favorite sneakers!

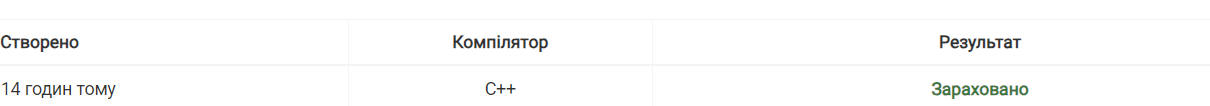
Підпис та № до блоку з виконанням та тестуванням програми

Час затрачений на виконання завдання 3 години

Завдання №6: Practice# programming: Self Practice Task 1

10 9 8 7 6

WIN



Підпис та № до блоку з виконанням та тестуванням програми

Час затрачений на виконання завдання 2 дні

# **Висновки:**

Я ознайомився з мовою програмування C/C++, створивши декілька програм з простою структурою. Також я здобув професійні навички в роботі з лінійними та розгалужними алгоритмами, умовними операторами, константантами і зміними. В результаті я створив звіт, який демонструє роботу, яку я виконав.