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. Змінні. Константи. Типи даних. Розмір типів даних. Ввід вивід. Базові операції та вбудовані функції. Коментарі.»

***з дисципліни:*** «Основи програмування»

до:

ВНС Лабораторної Роботи № 1

Алготестер Лабораторної Роботи № 1

Практичних Робіт до блоку № 2

**Виконав(ла):**

Студент групи ШІ-11

Боднар Денис

**Тема:**

Лінійні алгоритми. Розгалужені алгоритми. Умовні та логічні оператори. Змінні. Константи. Типи даних. Розмір типів даних. Ввід вивід. Базові операції та вбудовані функції. Коментарі.

**Мета роботи:**

Навчитися ефективно будувати програмну логіку та працювати з різними типами дани, створювати чіткі й структуровані програми, що вирішують різноманітні задачі з використанням стандартних інструментів програмування.

**Теоретичні відомості:**

1. Теоретичні відомості з переліком важливих тем:

* Тема №1: Використання систем числення в програмуванні.
* Тема № 2: Функції компілятора і етапи компіляції.
* Тема № 3: Змінні, Константи, Типи даних та їх розміри.
* Тема № 4:  Бібліотеки в C++.
* Тема № 5: Ввід та Вивід даних. Робота з різними типами даних у вводі/виводі.
* Тема № 6: Коментарі у Коді.  Види коментарів у С++.
* Тема № 7: Розгалужені алгоритми та Умовні Оператори.
* Тема № 8: Визначення та види логічних операторів.

1. Індивідуальний план опрацювання теорії:

Тема №1: Використання систем числення в програмуванні.

* + Джерела:
    - Стаття. <https://ua5.org/osnprog/28-ponjattja-sistemi-chislennja.html>
    - Відео. <https://www.youtube.com/watch?v=yGmVLDenVpE>

Що опрацьовано:

* + - * Позиційна і непозиційна системи числення.
      * Система числення, що використовується в комп’ютерах.
      * Вісімкова система числення в Linux (права доступу до файлів).
    - Статус: Ознайомлений

Тема №2: Функції компілятора і етапи компіляції.

* + Джерела:
    - Стаття. <https://coworkingclub.com.ua/uk/scho-take-kompilyator/>
    - Відео. <https://coworkingclub.com.ua/uk/scho-take-kompilyator/>

Що опрацьовано:

* + - * Компілятор та інтерпретатор – різниця.
      * Шість фаз компілятора це 1) лексичний аналіз 2) синтаксичний аналіз 3) семантичний аналіз 4) проміжний генератор коду 5) оптимізатор коду 6) код Generator.
    - Статус: Ознайомлений

Тема №3: Змінні, Константи, Типи даних та їх розміри.

* + Джерела:
    - Стаття .<https://www.guru99.com/uk/c-variable-datatypes-constants.html>
    - Відео. <https://acode.com.ua/urok-36-typy-danyh-z-plavayuchoyu-krapkoyu-float-double-i-long-double/>

Що опрацьовано:

* + - * Чотири типи даних, які зазвичай використовуються
      * Типи констант (перед вісімковою 0, перед шістнадцятковою 0X, символьні в одиничних лапках, рядкові в подвійних лапках).
      * Оголошення констант.
      * Float – 6-9 значень точності(частіше 7), double – 15-18 цифр(частіше 16), long double: 15, 18 чи 33.
    - Статус: Ознайомлений

Тема №4: Бібліотеки в C++.

* + Джерела:
    - Стаття. [клац](https://uk.wikipedia.org/wiki/%D0%A1%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D0%BD%D0%B0_%D0%B1%D1%96%D0%B1%D0%BB%D1%96%D0%BE%D1%82%D0%B5%D0%BA%D0%B0_C%2B%2B)
    - Відео. <https://www.youtube.com/watch?v=m-WJikuZGuU>

Що опрацьовано:

* + - * Стандартна бібліотека C — кожний заголовок стандартної бібліотеки C включено у стандартну бібліотеку C++ під іншою назвою: вилучено закінчення .h і додано літеру 'c' на початку.
      * Наприклад, Сmath та math.h, якщо написати math.h, то використати змінну pow вже не можна, адже існує така функція, з Сmath таких проблем немає. Усі функції з cmath знаходяться у просторі імен std. Тому, щоб використовувати функцію pow, потрібно вказати, що це std::pow. Тому, можна використовувати ім'я pow як змінну в іншому контексті без конфліктів, оскільки компілятор знає, що є різниця між глобальною змінною і функцією з простору імен std.
    - Статус: Ознайомлений

Тема №5: Ввід та Вивід даних. Робота з різними типами даних у вводі/виводі.

* + Джерела:
    - Стаття. <https://acode.com.ua/urok-215-potoky-vvodu-i-vyvodu/>
    - Відео. <https://www.youtube.com/watch?app=desktop&v=OAeYnm1uzS0>

Що опрацьовано:

* + - * Класи istream(Оператор вилучення >> значень з потоку), ostream (Оператор вставки <<  значень в потік) , iostream.
      * Потік вводу та виводу.
      * Специфікатори типів даних (%c - одиничний символ, %s -рядок, %d та %і десяткове ціле число, %p - адреса (або вказівник), %f - число з рухомою комою і тд).
      * Scanf та printf структура. Scanf("%d %f", &a, &b), printf("a: %d, b: %.2f\n", a, b);
    - Статус: Ознайомлений

Тема №6: Коментарі у Коді.  Види коментарів у С++.

* + Джерела:
    - Стаття. <https://w3schoolsua.github.io/c/c_comments.html#google_vignette>

Що опрацьовано:

* + - * Однорядкові коментарі починаються з двох скісних рисок (//).
      * Багаторядкові коментарі починаються з /\* і закінчуються \*/
    - Статус: Ознайомлений

Тема №7: Розгалужені алгоритми та Умовні Оператори.

* + Джерела:
    - Стаття. <https://acode.com.ua/urok-67-operatory-umovnogo-rozgaluzhennya-if-else/>

Що опрацьовано:

* + - * Умовні розгалуження if/else.
      * Оператор switch, лейбли case.
      * switch і fall-through, switch і оператор break
      * Тарнарний оператор.
    - Статус: Ознайомлений

Тема №8: Визначення та види логічних операторів.

* + Джерела:
    - Стаття. <https://acode.com.ua/urok-46-logichni-operatory-i-abo-ne/>

Що опрацьовано:

* + - * Використання операторів І/АБО/НЕ
    - Статус: Ознайомлений

**Виконання роботи:**

**VNS Lab 1 Task 1 & VNS Lab 1 Task 1:**

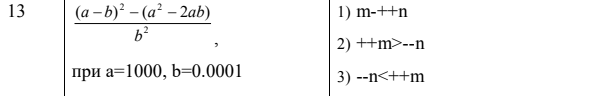
1. Обчислити значення виразу при різних дійсних типах даних (float й double).

Обчислення варто виконувати з використанням проміжних змінних.

Порівняти й пояснити отримані результати.

2. Обчислити значення виразів. Пояснити отримані результати.

№1 №2



**Завдання №3** **– Algotester lab 1 Variant 3**

**Задача:**

Персонажу по одному дають сторони 5 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..a5a1..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

**Constraints**

≤ai≤

**Завдання №4 - Class Practice Work - Особистий порадник**

**Задача**

Ви створюєте простий порадник щодо погоди. Користувач вводить поточні погодні умови, а програма видає рекомендації щодо активності на основі погоди.

**Можливі варіанти погоди:**

* sunny;
* rainy;
* cloudy;
* snowy;
* windy;

**Мета Задачі**

Навчитися користуватися операторами галуження для структурування логіки програм.

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.

**Завдання №5 - Self Practice Work – Algotester Щасливий результат**

**Задача:**

Після завершення основного туру олімпіади з програмування Зеник отримав невеличкий клаптик паперу, на якому було надруковане число xx — кількість балів, що набрав Зеник. Зауважте, що згідно з кращими традиціями олімпіади з програмування, кількість балів Зеника не може бути нульовою чи від’ємною.

Помітивши не дуже щасливе обличчя Зеника, Марічка нагадала йому про щасливі цифри. Як ви вже напевно знаєте, щасливими вважають цифри 4 та 7. Марічка запевнила Зеника, що найкращим є не найбільший результат, а той, десятковий запис якого містить найбільше щасливих цифр.

Вам необхідно допомогти юному учаснику олімпіади з програмування та порахувати кількість щасливих цифр у його результаті.

**Вхідні дані**

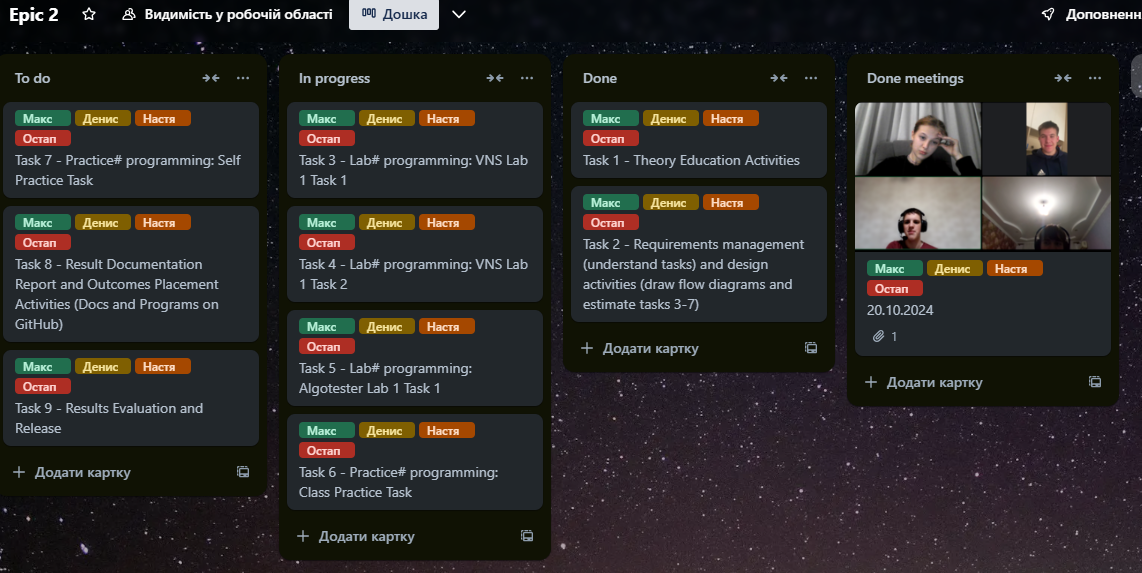
У єдиному рядку задано одне ціле число xx — результат Зеника.

**Вихідні дані**

У єдиному рядку виведіть одне ціле число — кількість щасливих цифр у десятковому записі xx.

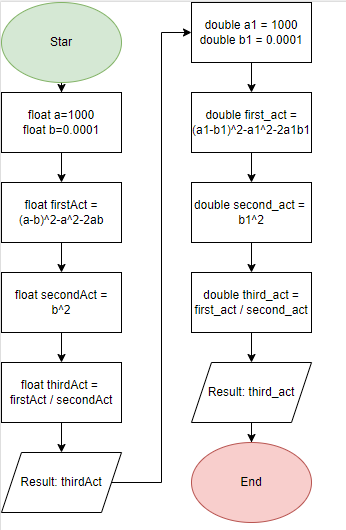
**Обмеження** 1 ≤ x ≤109

**Дошка Trello**

****

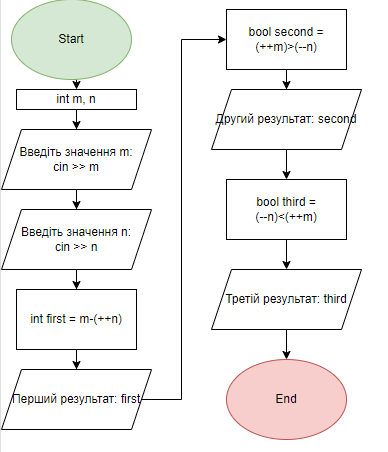
**Діаграми**

**Завдання №1 - VNS Lab 1 - Task 1- Variant 13**

****

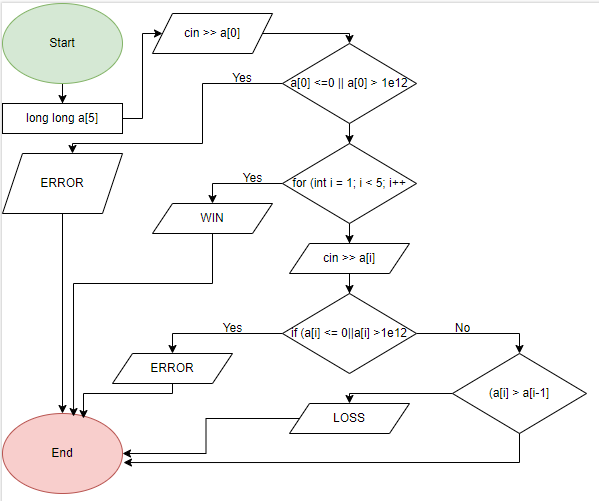
Планований час: 20хв

**Завдання №2 - VNS Lab 1 - Task 2 - Variant 13**



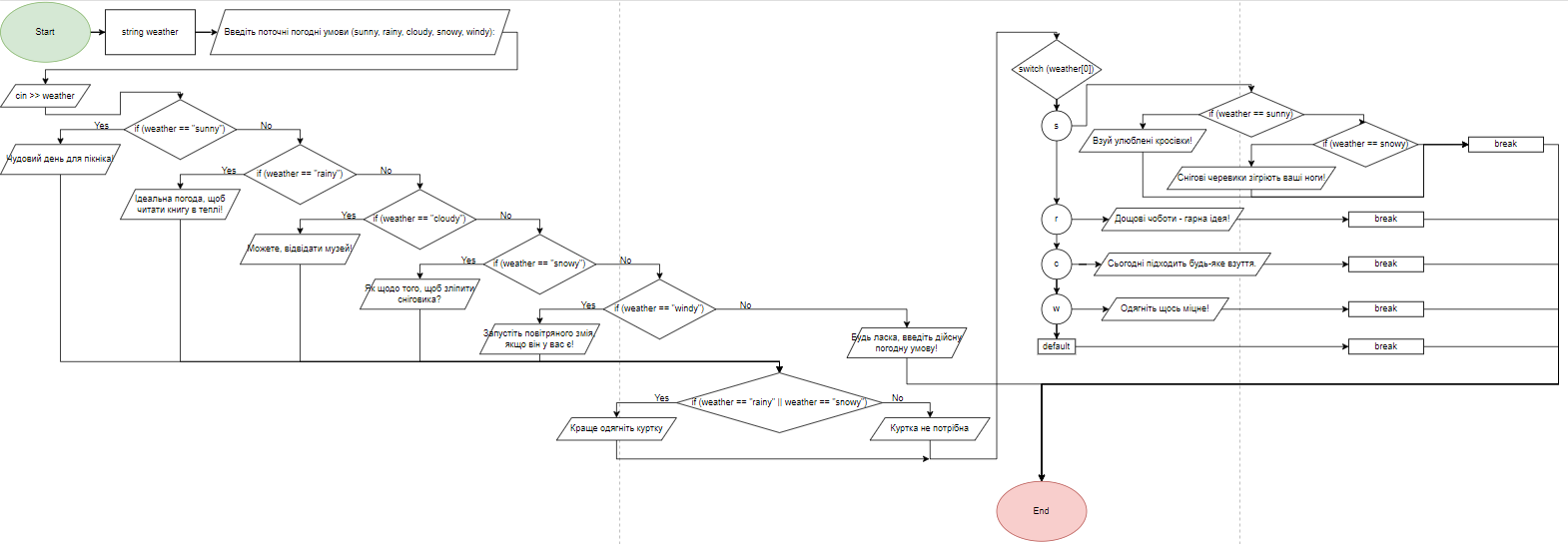
Планований час: 20хв

**Завдання №3 – Algotester lab 1 Variant 3**

****

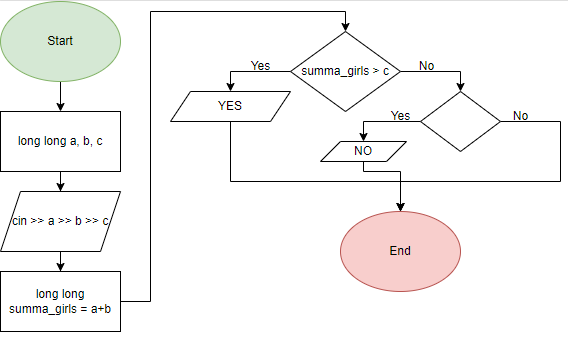
Планований час: 30хв

**Завдання №4 - Class Practice Work - Особистий порадник**



Планований час: 1 година

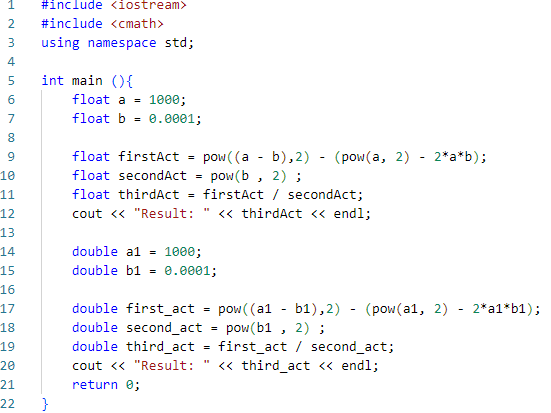
**Завдання №5 - Self Practice Work – Algotester Апельсини**

****

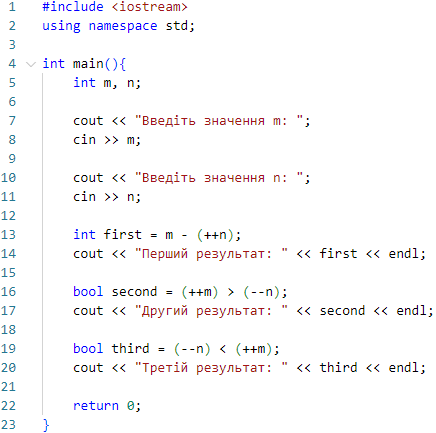
Планований час: 30хв

**Коди програм**

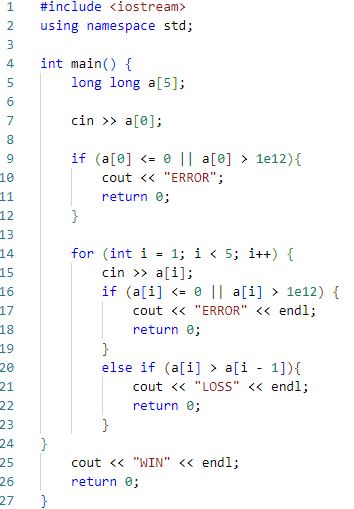
**Завдання №1**

****

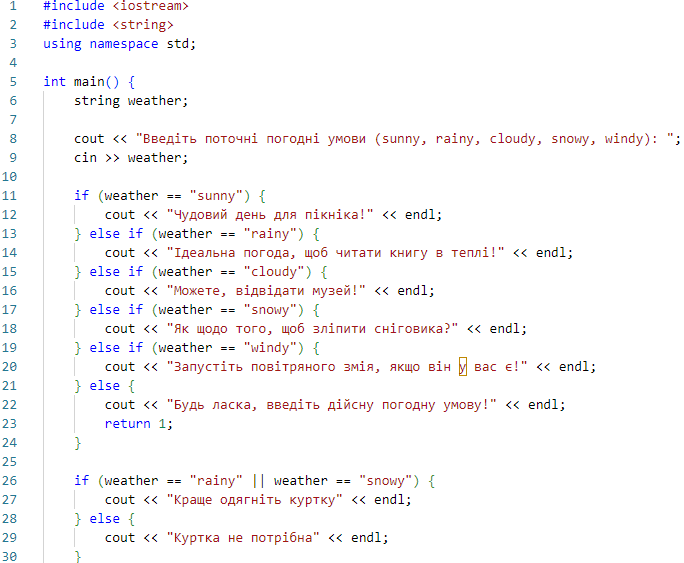
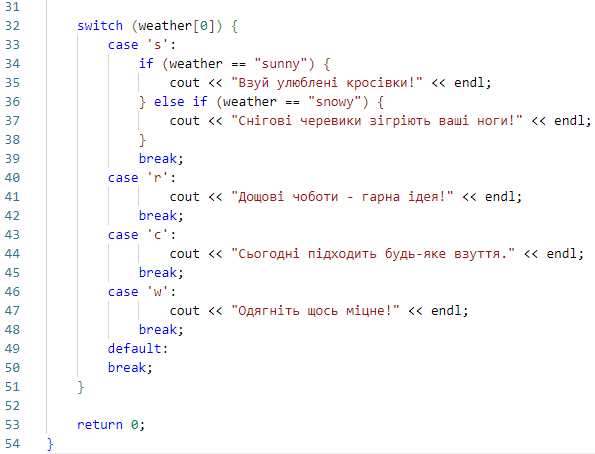
**Завдання №2**

****

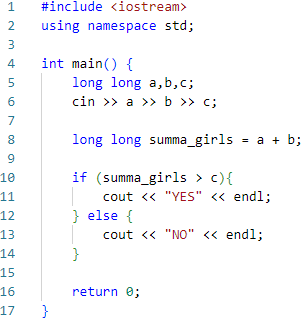
**Завдання №3**

****

**Завдання №4**

**** ****

**Завдання №5**

****

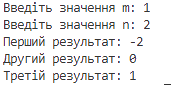
**Результати виконання завдань**

**Завдання №1  VNS Lab 1 Task 1**



Фактично затрачений час: 15хв

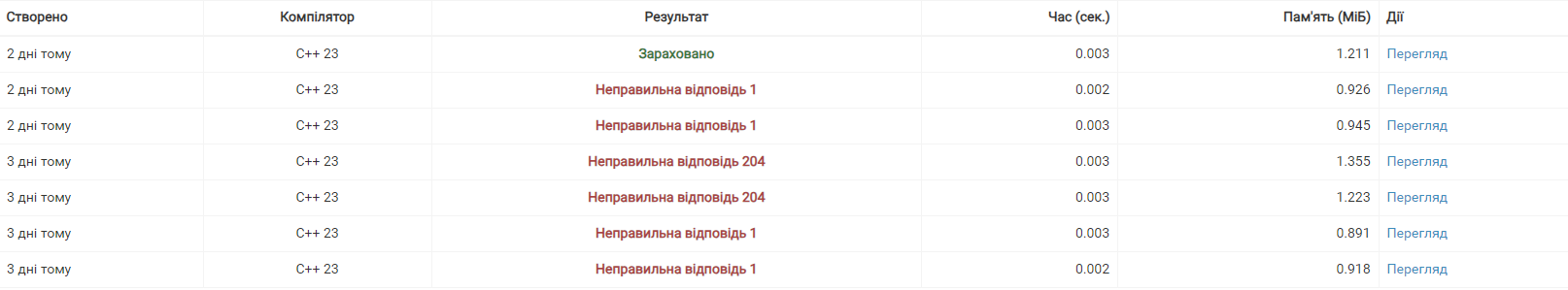
**Завдання №2 VNS Lab 1 Task 2**

****

Фактично затрачений час: 20хв

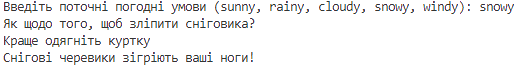
**Завдання №3 Algotester Lab 1 Task 1**

****

****

Фактично витрачений час: 2-3 години

**Завдання №4 Class Practice Task**

****

Фактично затрачений час: 45хв

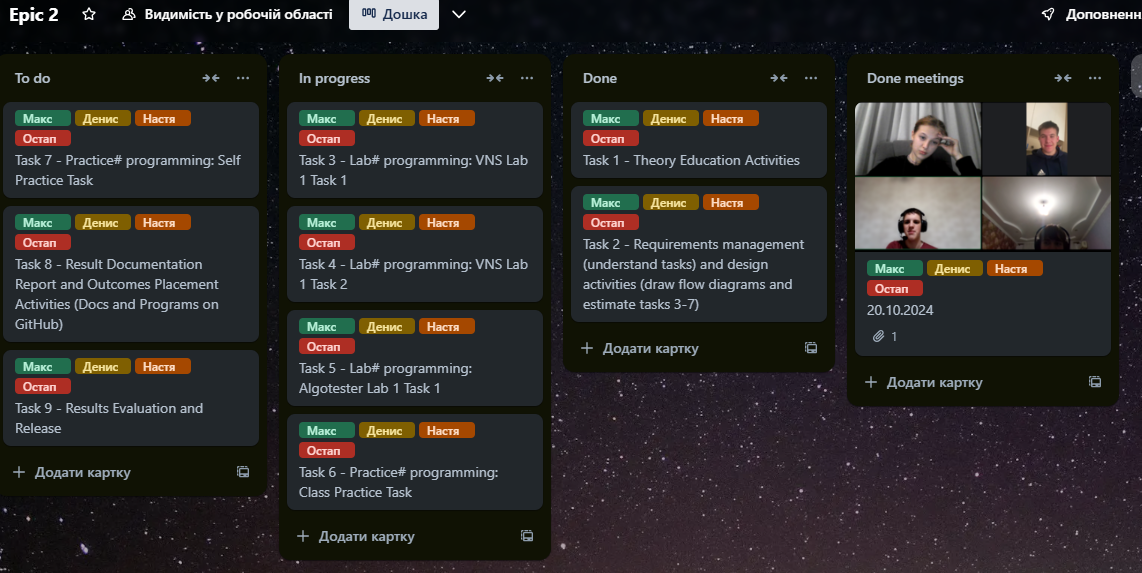
**Завдання №5 Self Practice Task**





Фактично затрачений час: 15хв

**Кооперація з командою**

****

Зустрічалися з командою в discord та обговорили всі деталі по епіку 2

**Висновок:**

Під час виконання другого епіку я ознайомився з основними етапами компіляції та роботою препроцесора, що дало мені краще розуміння того, як працює компілятор та які функції він виконує для оптимізації коду. Я засвоїв принципи оголошення змінних і констант, а також вибір відповідних типів даних, що є важливими для ефективної роботи з пам'яттю. Зрозумів мету використання стандартних бібліотек, таких як iostream та cmath. Я оволодів навичками форматування виводу, написання лінійних алгоритмів і використання умовних операторів для побудови складніших програм. А також на практиці відпрацював структуру розгалуження, написавши декілька програм з їх використанням.