Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра систем штучного інтелекту



**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-12

Бортник Тарас Петрович

**Тема роботи:**

# **Вивчення лінійних та розгалужених алгоритмів, умовних операторів, робота зі змінними та константами, операції введення-виведення у мові програмування C++.**

# **Мета роботи:**

# **Ознайомлення з основами програмування на C++ та використання умов та операцій введення-виведення для розробки простих програм.**

# **Теоретичні відомості:**

1. Теоретичні відомості з переліком важливих тем:

* Тема №1: Лінійні алгоритми
* Тема №2: Умовні оператори
* Тема №3: Розгалужені алгоритми

1. Індивідуальний план опрацювання теорії:

* Тема №1: Лінійні алгоритми
  + Джерела Інформації
    - Відео.
    - <https://www.youtube.com/watch?v=RIU2y1EkS1U>
    - Стаття
    - http://cpp.dp.ua/ponyattya-algoritmu/
  + Що опрацьовано:
    - Вивчення та впровадження послідовних алгоритмів для вирішення завдань в програмуванні.
    - Освоєння основ лінійних алгоритмів: послідовних кроків у програмуванні для ефективного вирішення завдань.
  + Статус: Ознайомлений
  + Початок опрацювання теми: 14.10.2023
  + Звершення опрацювання теми: 14.10.2023
* Тема №2: Умовні оператори.
  + Джерела Інформації:.
    - Відео.
    - <https://www.youtube.com/watch?v=r7Gg_B_FXDs>
    - Стаття.
    - <https://acode.com.ua/urok-67-operatory-umovnogo-rozgaluzhennya-if-else/>
    - <https://www.bestprog.net/uk/2017/08/02/conditional-jump-operator-if-2_ua/>
  + Що опрацьовано:
    - Вивчення умовних операторів розвинуло мою здатність створювати гнучкі програми, реагуючи на різні умови, що полегшило написання ефективного та адаптивного коду.
    - Вивчення умовних операторів розширило мої можливості у написанні програм, які інтелігентно реагують на різні умови, забезпечуючи гнучкість та точність у виконанні завдань.
  + Статус: Ознайомлений
  + Початок опрацювання теми: 14.10.2023
  + Звершення опрацювання теми: 14.10.2023
* Тема №3: Розгалужені алгоритми
  + Джерела Інформації:
    - Стаття
    - <https://disted.edu.vn.ua/courses/learn/10348>
  + Що опрацьовано:
    - Опрацювання розгалужених алгоритмів навчило створювати гнучкі програми, які адаптують свою дію до різних умов, роблячи код більш ефективним та адаптивним.
    - Коментар 2
  + Статус: Ознайомлений
  + Початок опрацювання теми: 15.10.2023
  + Звершення опрацювання теми: 15.10.2023

# **Виконання роботи:**

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

Завдання №1 VNS lab 1 task 1

* Варіант завдання : 22
* Деталі завдання
* Цей код виконує обчислення виразу, використовуючи змінні типу float. Основна ідея полягає в тому, що спочатку обчислюється чисельник і знаменник, а потім результат обчислення виразу записується в змінну result і виводиться на екран.

Зміна типу даних на double дозволяє використовувати більше бітів для зберігання значень і може забезпечити більшу точність при обчисленнях.

* Важливі деталі для врахування в імплементації програми
* Важливо враховувати точність числових типів, особливо у випадку типу `float`, щоб уникнути втрат точності при роботі з великими або дуже малими значеннями. Користуйтеся функціями з бібліотеки `<cmath>` для виконання математичних операцій. Перевіряйте вирази уважно, щоб уникнути помилок та підвищити продуктивність програми.

Завдання №2 VNS lab 1 task 2

* Варіант завдання: 22
* Деталі завдання:
* Цей код використовує бібліотеки <iostream> та <cmath>. Він приймає введені користувачем числа m і n та виводить результати трьох виразів: добуток (n+1) \* (m+1), порівняння m і n після постфіксного інкременту m, та порівняння n і m після постфіксного інкременту n. Результати порівнянь виводяться у вигляді логічних значень (true або false). Програма використовує інкременти, що може впливати на значення змінних.
* Важливі деталі для врахування в імплементації програми
* Важливі деталі включають використання бібліотек <iostream> та <cmath>, введення користувача для m і n, префіксний інкремент та порівняння результатів у форматі логічних значень.

Завдання №3 Algotester lab1

* Варіант завдання: 1
* Деталі завдання
* Цей код отримує введення щодо початкового часу (hbegin, mbegin) та трьох інших часових відрізків (h1, m1, h2, m2, h3, m3). Використовується цикл for та умовні оператори для визначення можливості відняти часові відрізки від початкового часу. Результат, "YES" або "NO", виводиться в залежності від умов.
* Важливі деталі для врахування в імплементації програми
* Цей код отримує введення щодо часу початку та трьох інших часових відрізків, використовуючи цикл та умовні оператори для визначення можливості відняти відрізки від початкового часу. Результат "YES" виводиться, якщо це можливо, інакше виводиться "NO"

Завдання №4 Class practice

* Варіант завдання 1
* Деталі завдання :
* Цей код створює простий порадник щодо погоди. Користувач вводить поточні погодні умови, і програма виводить рекомендації щодо одягу та активностей, використовуючи оператори умови та перeключeння.
* Важливі деталі для врахування в імплементації програми:
* Програма враховує тип погоди, щоб надати рекомендації з одягу та активностей. Використовуються оператори умови (if-else) для визначення, чи потрібно носити куртку, та switch-case для різних типів погоди, щоб вказати рекомендоване взуття. Код також перевіряє правильність введення погоди.

Завдання №5 Self-practice

Варіант завдання:name of task (Депутатські гроші)

* Деталі завдання :
* Цей код призначений для обчислення мінімальної кількості банкнот та монет, які складають задану суму грошей (введену користувачем). Використовуються вартості різних банкнот та монет, і програма визначає, скільки кожного типу потрібно для утворення введеної суми. Результат виводиться у вигляді кількості використаних банкнот та монет.
* Важливі деталі для врахування в імплементації програми:
* Програма приймає від користувача суму грошей та визначає мінімальну кількість банкнот та монет для утворення цієї суми. Використовується масив вартостей грошових одиниць, цикли для перевірки кожного типу грошей та цикл while для обчислення кількості. Результат виводиться у вигляді кількості використаних банкнот та монет.

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

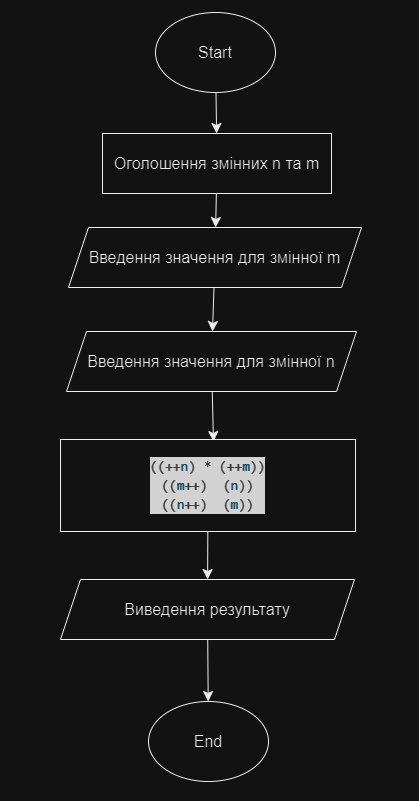
Програма №1 VNS lab 1 task 1



Рис №1 Блок-схема до 1 задачі

* Планований час на реалізацію 20 хв
* Важливі деталі для врахування в імплементації: Будьте уважні з дужками у вашому коді та уникайте написання всієї формули в одному рядку, щоб уникнути помилок. Використовуйте відступи та правильний порядок розташування дужок для поліпшення читабельності коду.

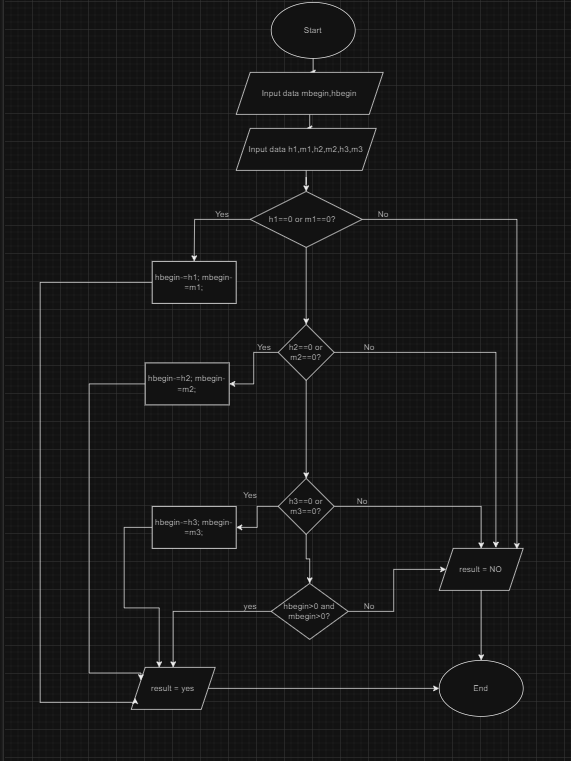
Програма №2 VNS lab 1 task 2



Рис№2 Блок-схема до другої задачі

* Планований час на реалізацію 20 хв
* Важливі деталі для врахування в імплементації: Важливо уважно вивчати кожен етап виконання готового коду та перевіряти результати виводу на кожному етапі операцій.

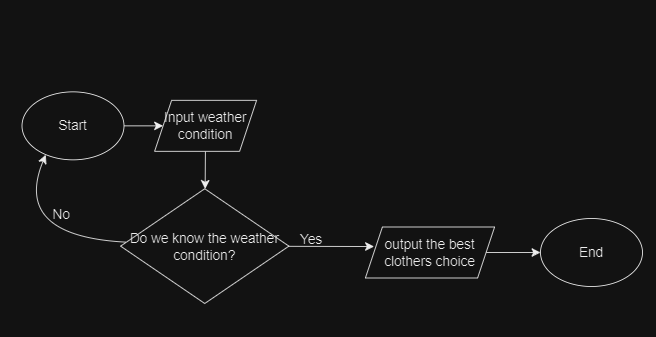
Програма №3 Algotester lab1



Рис№3 Блок-схема до третьої задачі

* Планований час на реалізацію 1 година

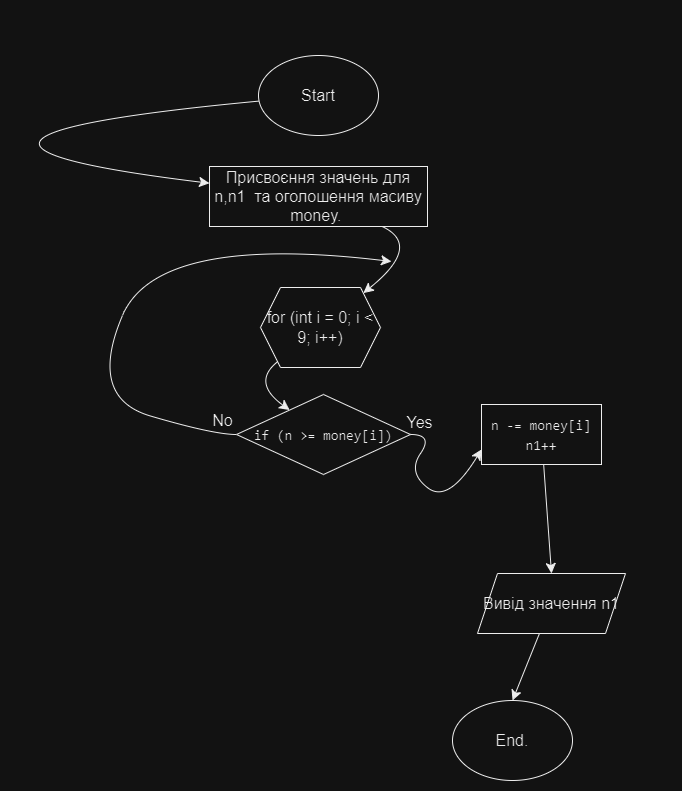
Програма №4 Class practice



Рис№4 Блок-схема до Class practice task

-Планований час на реалізацію 30 хвилин

Програма №5Self-practice



Рисунок№5 Блок-схема до Self practice task

Планований час на реалізацію 25 хвилин.

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

**Завдання №1 Деталі по програмі:**

Програма обчислює та виводить результат виразу, використовуючи тип float для чисел. Важливо враховувати точність числових типів та уникати втрат точності при роботі з числовими значеннями.

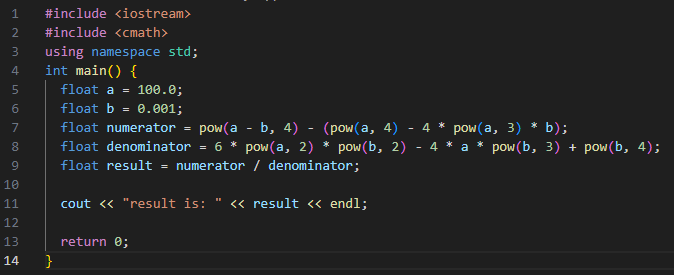


Рисунок №6 Малюнок до коду vnslab1task1

[Посилання на код у ПР](https://github.com/artificial-intelligence-department/ai_programming_playground/pull/529/commits/ca1c511dadc961fd139193358182a06330f11d51#diff-008afcf3f430e8e3a54680fc026fe337e66f156d32249f588add26de9785ac5f)

**Завдання №2 Деталі по програмі:**

Програма приймає від користувача числа n і m, обчислює та виводить результати трьох виразів, використовуючи операції інкремента та порівняння.

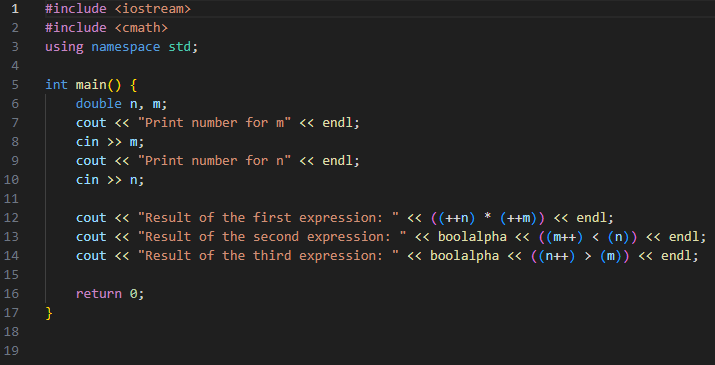


Рисунок №7 Малюнок до коду vnslab1task2

[Посилання на код у ПР](https://github.com/artificial-intelligence-department/ai_programming_playground/pull/529/commits/ca1c511dadc961fd139193358182a06330f11d51#diff-6d71f0301ed791ab67f94da6eb9e8a8f853451b637d53fc758c371ec5cf785fc)

**Завдання №3 Деталі по програмі:**

Програма приймає вхідні дані для часу початку та трьох інших моментів часу. Вона перевіряє, чи можливо відняти час кожного із трьох моментів від початкового, і виводить "YES", якщо це можливо, та "NO" у протилежному випадку.

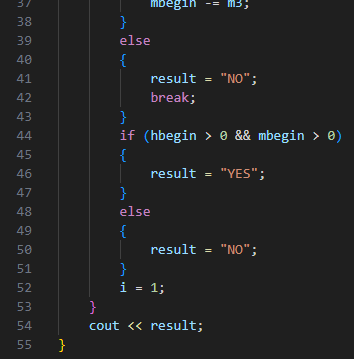
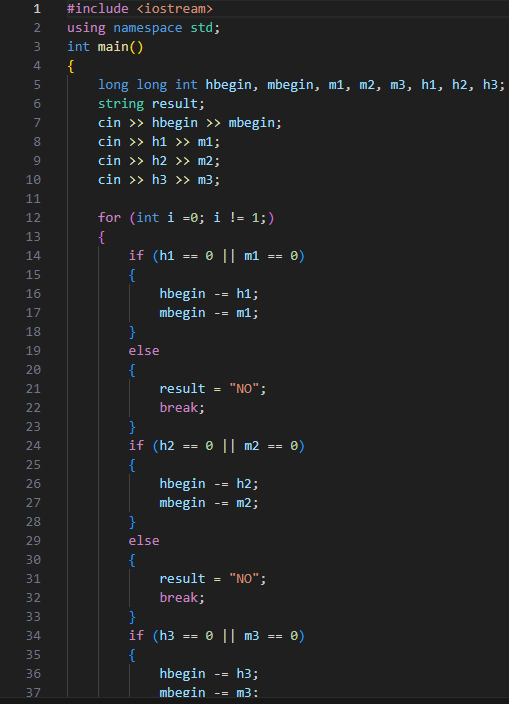


Рис №8 Програма для перевірки можливості відняття трьох часових інтервалів від початкового часу

[Посилання на програму у ПР](https://github.com/artificial-intelligence-department/ai_programming_playground/pull/529/commits/ca1c511dadc961fd139193358182a06330f11d51#diff-a5a510101078af0fb0d32c1438be8cd1a73e7f7d9ba59910c454b801611fe436)

**Завдання №4 Деталі по програмі:**

Програма надає поради щодо одягу та активностей в залежності від введеної користувачем погоди. Використовує умовний оператор та конструкції switch.

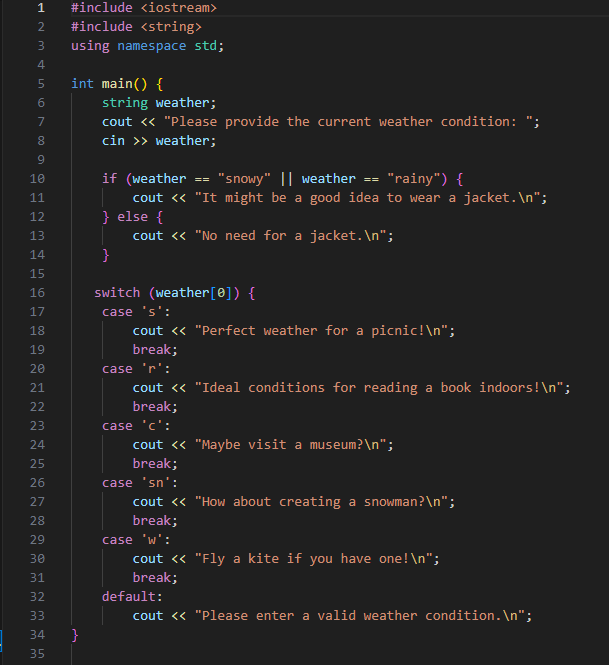


Рисунок №9 Програма для надання порад з одягу та занять в залежності від погодних умов

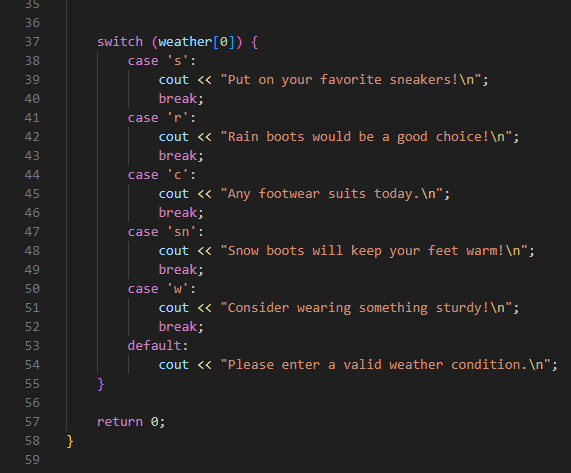
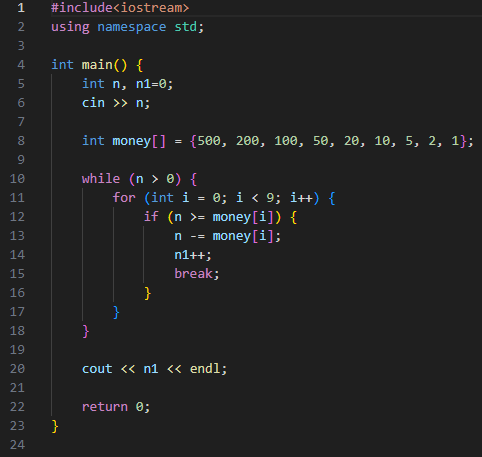


Рисунок №10 Програма для надання порад з одягу та занять в залежності від погодних умов

[Посилання на код у ПР](https://github.com/artificial-intelligence-department/ai_programming_playground/pull/529/commits/ca1c511dadc961fd139193358182a06330f11d51#diff-4b2eb50f34941979609e297b18561afa9336aa0e5fa52efe4695d55ae5624635)

**Завдання №5 Деталі по програмі:**

Програма отримує від користувача число та розкладає його на кількість монет, використовуючи сталі номінали. Використовується цикл while та масив money.



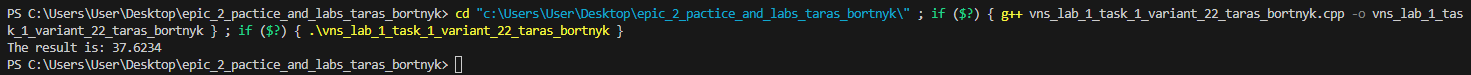
Рисунок№11 Програма для розкладання суми на монети за фіксованими номіналами

[Посилання на код у ПР](https://github.com/artificial-intelligence-department/ai_programming_playground/pull/529/commits/ca1c511dadc961fd139193358182a06330f11d51#diff-9f8b29f66137f90fcfcecba585fb8f68cf424c66e0292b1f00285af309ed29a5)

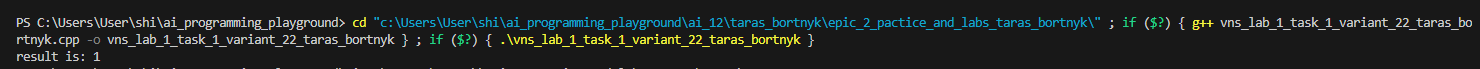
## **4. Результати виконання завдань, тестування та фактично затрачений час:**

Завдання №1 Деталі по виконанню і тестуванню програми

Програма успішно виконалася і повернула очікуваний результат.



Рисунок№12 вивід double

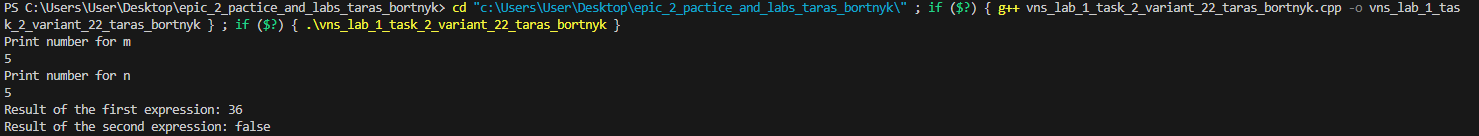
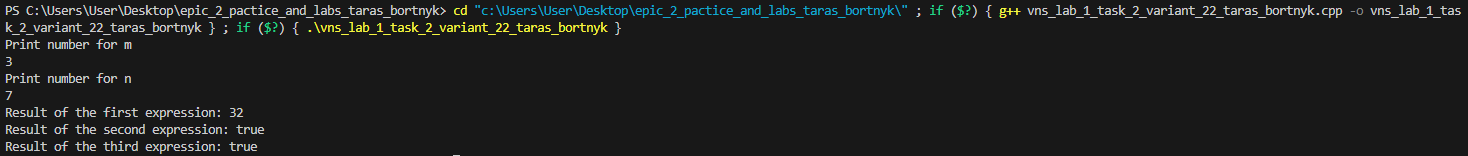


Рисунок№13 вивід float

Час затрачений на виконання: 20 хв

Завдання №2 Деталі по виконанню і тестуванню програми

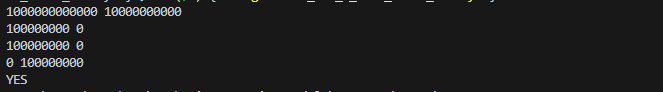
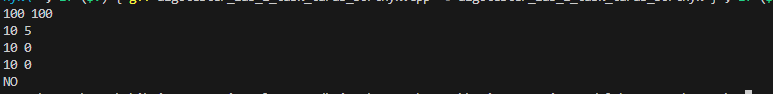
Програма працює коректно



Рисунок№14 Вивід консолі vnslab1task2

Час затрачений на реалізацію: 20 хв

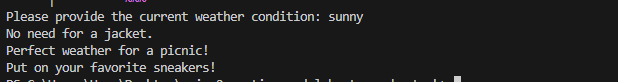
Завдання №4 Деталі по виконанню і тестуванню програми



Рисунок№14 Вивід консолі algotester lab1

Час затрачений на реалізацію: 1 година 30 хвилин

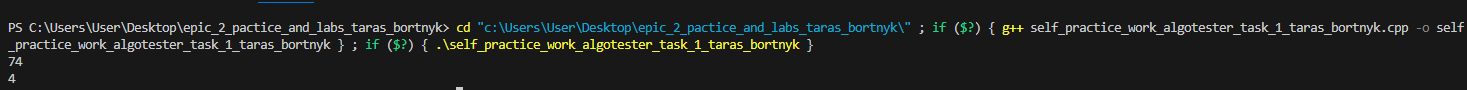
Завдання №2 Деталі по виконанню і тестуванню програми



Рисунок№15 Вивід консолі class-practice task

Час затрачений на реалізацію:45 хвилин

Завдання №4 Деталі по виконанню і тестуванню програми



Рисунок№16 Вивід результату у консоль

Час затрачений на виконання: 25 хвилин

# **Висновки:**

У цій темі ми ознайомилися з умовними операторами, такими як if/else/else if/switch, і розширили свої знання з бібліотеки <cmath>. Здобули практичні навички вирішення завдань, пов'язаних з математикою. Наприклад, в одному із завдань була написана функція для обчислення факторіалу. Додатково розвинули розуміння роботи з змінними та константами. Отримали досвід використання масивів, виконували операції, такі як додавання та видалення елементів із масиву, а також створення нових масивів з існуючих елементів.