Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра систем штучного інтелекту



**Звіт**

про виконання

**Лабораторних та практичних робіт**

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

***з розділу***: «Лінійні та розгалужені алгоритми. Умовні оператори.

Константи, змінні»

***Виконав:***

студент групи ШІ-12

Капустяк Роман Миколайович

# **Тема роботи:**

Знайомство із С++. Виконання програм простої структури із лінійними і розгалуженими алгоритмами. Використання основних операторів мови С++.

# **Мета роботи:**

Ознайомитись із мовою програмування С++ шляхом створення кількох програм та складання для них лінійних і розгалужених алгоритмів. Навчитись використовувати основні оператори мови С++.

# **Теоретичні відомості:**

1. Теоретичні відомості з переліком важливих тем:

* Тема №1: Структура програми.
* Тема №2: Константи й змінні.
* Тема №3: Операції та вирази.
* Тема №4: Ввід і вивід.
* Тема №5: Оператори вибору.

1. Індивідуальний план опрацювання теорії:

* Тема №1: Структура програми.
  + Джерела Інформації
    - Стаття.

https://www.geeksforgeeks.org/structure-of-cpp-program/Що опрацьовано:

* + - Ознайомився із структурою програми на прикладі мови C++.
  + Що опрацьовано:
    - Дізнався складові структури, їх суть та порядок запису.
  + Статус: Ознайомлений.
  + Початок опрацювання теми: 30.10.2023
  + Звершення опрацювання теми: 30.10.2023
* Тема №2: Константи й змінні.
  + Джерела Інформації:
    - Стаття.

https://www.programiz.com/cpp-programming/variables-literals

* + Що опрацьовано:
    - Ознайомився із типами даних, способами оголошення змінних та використанням констант.
  + Статус: Ознайомлений.
  + Початок опрацювання теми: 30.10.2023
  + Звершення опрацювання теми: 30.10.2023
* Тема №3 Операції та вирази.
  + Джерела Інформації:
    - Стаття.

https://www.javatpoint.com/cpp-expression

https://www.w3schools.com/cpp/cpp\_operators.asp

* + Що опрацьовано:
    - Ознайомився із принципами роботи із вказівниками, цілочисельними виразами, десятковими, порівняльними, логічними та деякими побітовими виразами.
    - Також ознайомився із основними математичними операторами.
  + Статус: Ознайомлений.
  + Початок опрацювання теми: 30.10.2023
  + Звершення опрацювання теми: 30.10.2023
* Тема №4: Ввід і вивід.
  + Джерела Інформації:
    - Стаття.

https://www.programiz.com/cpp-programming/input-output

https://www.w3schools.com/cpp/cpp\_strings\_input.asp

* + Що опрацьовано:
    - Ознайомився із стандартними потоками cin і cout, що забезпечують ввід і вивід даних консолі.
    - Також дізнався про способи отримання даних з консолі, що можуть містити в собі пробіли.
  + Статус: Ознайомлений.
  + Початок опрацювання теми: 30.10.2023
  + Звершення опрацювання теми: 30.10.2023
* Тема №5 Оператори вибору.
  + Джерела Інформації:
    - Стаття.

https://www.programiz.com/cpp-programming/if-else

* + - Ознайомився із операторами переходу if та if…else.
  + Статус: Ознайомлений.
  + Початок опрацювання теми: 30.10.2023
  + Звершення опрацювання теми: 30.10.2023

# **Виконання роботи:**

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

Завдання №1 VNS Lab 1 - Task 1-1

* Варіант завдання 24
* Деталі завдання:

Обчислити значення заданого виразу.

* Важливі деталі для врахування в імплементації програми:

Розглянути випадки при різних дійсних типах початкових даних (float й double).

Завдання №2 VNS Lab 1 - Task 1-2

* Варіант завдання 24
* Деталі завдання:

Обчислити значення заданих виразів.

Завдання №3 VNS Lab 2

* Варіант завдання 2
* Деталі завдання:

Знайти суму зазначених елементів.

* Важливі деталі для врахування в імплементації програми:

Потрібно використати оператор циклу.

Завдання №4 Algotester Lab 1

* Варіант завдання 3
* Деталі завдання:

Персонажу по одному дають сторони 5 кубiв a(1..5), з яких вiн будує пiрамiду.

Коли вiн отримує куб з ребром a(i) - вiн його ставить на iснуючий, перший ставить на пiдлогу (вона безмежна). Якщо в якийсь момент об’єм куба у руцi (який будуть ставити) буде бiльший нiж у куба на вершинi пiрамiди - персонаж програє i гра закiнчується. Розмiр усiх наступних кубiв пiсля програшу не враховується. Тобто якщо a(i−1) < a(i) - це програш. Завдання - сказати як закiнчиться гра.

* Важливі деталі для врахування в імплементації програми:

Вхiднi данi:

5 цiлих чисел a(1)..a(5) - сторони кубiв.

Вихiднi данi:

Iснуючi варiанти:

LOSS - якщо персонаж не зможе поставити куб.

WIN - якщо персонаж зможе поставити усi куби.

ERROR - якщо сторона куба буде мати неможливу довжину, тобто a(i) <= 0.

Завдання №5 Algotester Lab 2

* Варіант завдання 2
* Деталі завдання:

Дано масив r розмiром N і 3 цiлих числа. Спочатку необхідно видалити з масиву цi 3 числа. Пiсля цього перетворити цей масив у масив сум, розмiром N(new) − 1 (розмiр нового масиву пiсля видалення елементiв), який буде вiдображати суми сусiднiх елементiв нового масиву. Далi необхiдно вивести масив сум на екран.

* Важливі деталі для врахування в імплементації програми:

Вхiднi данi:

У першому рядку цiле число N - кiлькiсть чисел. У другому рядку масив r, який складається з N цiлих чисел. У третьому рядку 3 цiлих числа, a, b, c, якi треба видалити з масиву.

Вихiднi данi:

У першому рядку цiле число M - кiлькiсть чисел у масивi, який буде виведено. У наступному рядку M чисел - новий масив.

Завдання №6 Class Practice Work

* Деталі завдання:

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

* Важливі деталі для врахування в імплементації програми:

Запропонувати користувачеві ввести поточні погодні умови із заданого списку. Якщо користувач вводить будь-яку іншу умову, запропонувати йому ввести дійсну умову.

Завдання №6 Self Practice Work Наноробот

* Деталі завдання:
* Робот косить траву на прямокутному майданчику, починаючи з північно-західного кута. Сторони майданчика є паралельними до сторін світу. Траєкторія руху робота за формою нагадує спіраль — спочатку рух виконується у східному напрямку, тоді у південному, далі у західному, потім у північному і знову повторюється.  
  За один крок робот опрацьовує ділянку розміром 1 на 1 метр. Під час руху робот не перетинає меж майданчика і не потрапляє на вже опрацьовані ділянки, після останнього можливого кроку зупиняється. Потрібно визначити координати ділянки (по горизонталі і по вертикалі), у якій опиниться робот на заданому кроці. Початкова ділянка (на кроці 1) має координати 1 1.
* Важливі деталі для врахування в імплементації програми:   
  Вхідні дані:  
  У єдиному рядку задано три цілих числа розділені пробілами — ширина майданчика по горизонталі, довжина по вертикалі у метрах та номер кроку робота відповідно.  
  Вихідні дані:  
  У єдиному рядку вивести два невід’ємних цілих числа, розділені пробілом — координати по горизонталі та вертикалі для ділянки, де буде робот на введеному кроці.
* Якщо такий крок є неможливим за умовою задачі, вивести два числа через пробіл: -1 -1.

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

Програма №1 VNS Lab 1 - Task 1-1

* Блок-схема:

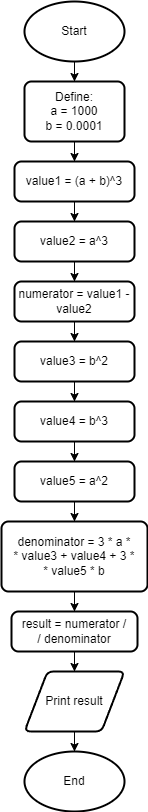


Рисунок 1: Блок-схема до програми №1

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

Обчислення варто виконувати з використанням проміжних змінних.

Програма №2 VNS Lab 1 - Task 1-2

* Блок-схема:

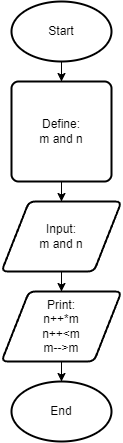


Рисунок 2: Блок-схема до програми №2

* Планований час на реалізацію 1 день.

Програма №3 VNS Lab 2

* Блок-схема:

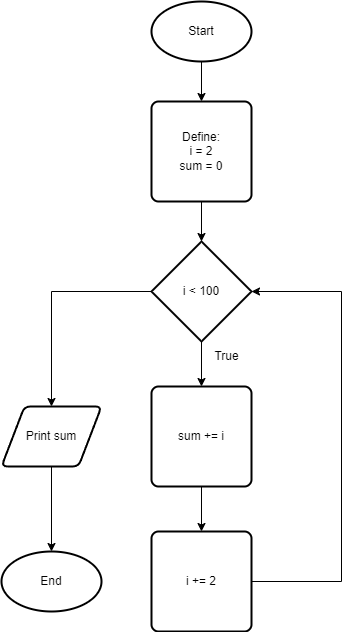


Рисунок 3: Блок-схема до програми №3

* Планований час на реалізацію 1 день.

Програма №4 Algotester Lab 1

* Блок-схема:

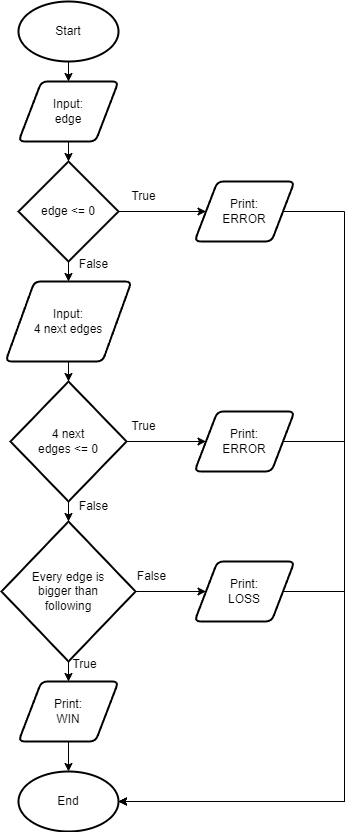


Рисунок 4: Блок-схема до програми №4

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

Потрібно використовувати тип даних long long через обмеження вхідних даних.

Програма №5 Algotester Lab 2

* Блок-схема:

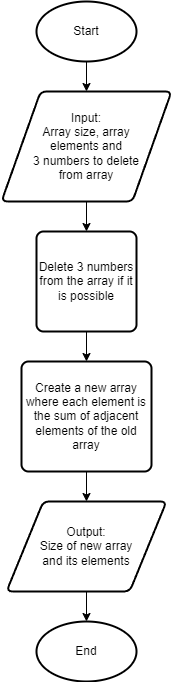


Рисунок 5: Блок-схема до програми №5

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

Вхідні дані подаються у вигляді цілих невід’ємних чисел.

Програма №6 Class Practice Work

* Блок-схема:

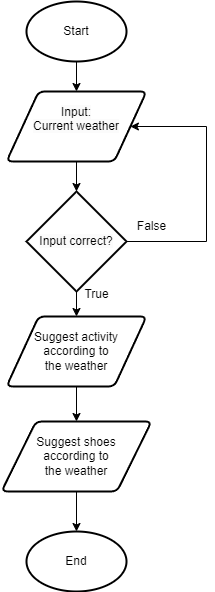


Рисунок 6: Блок-схема до програми №6

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

Використати 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.

Програма №7 Self Practice Work Наноробот

* Блок-схема:

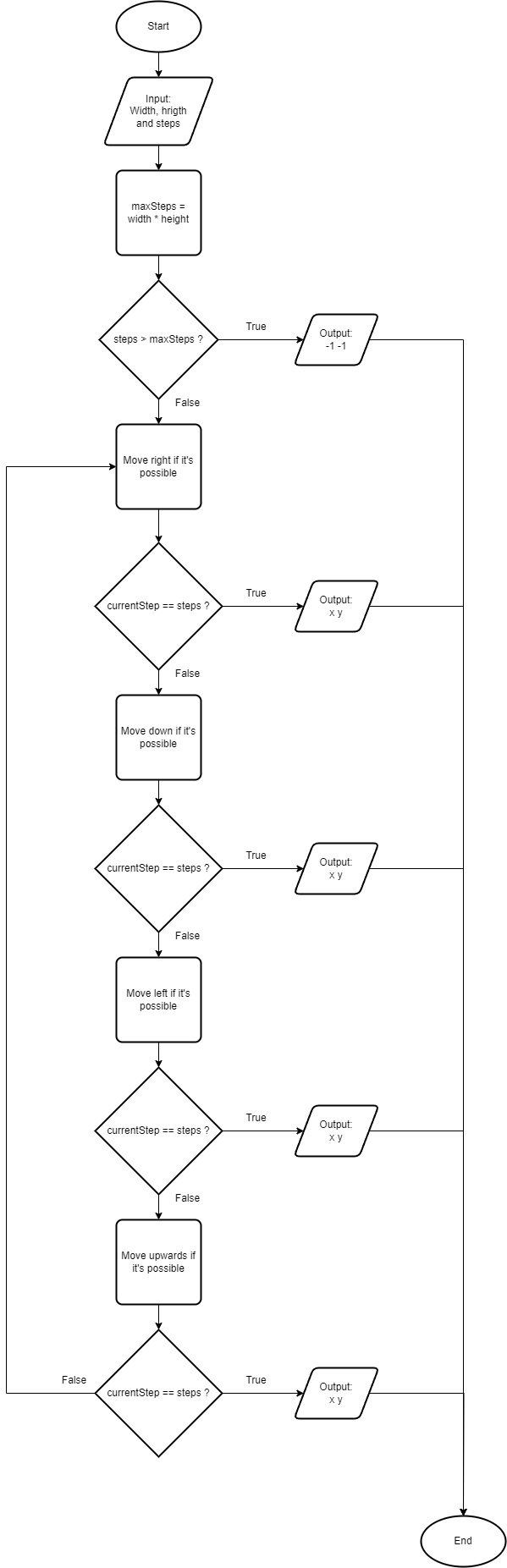


Рисунок 7: Блок-схема до програми №7

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

Вхідні дані подаються у вигляді цілих невід’ємних чисел, де перші два значення повинні бути більшими за 1.

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

* Додаткова конфігурація середовища не потрібна.

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

Завдання №1 VNS Lab 1 - Task 1-1

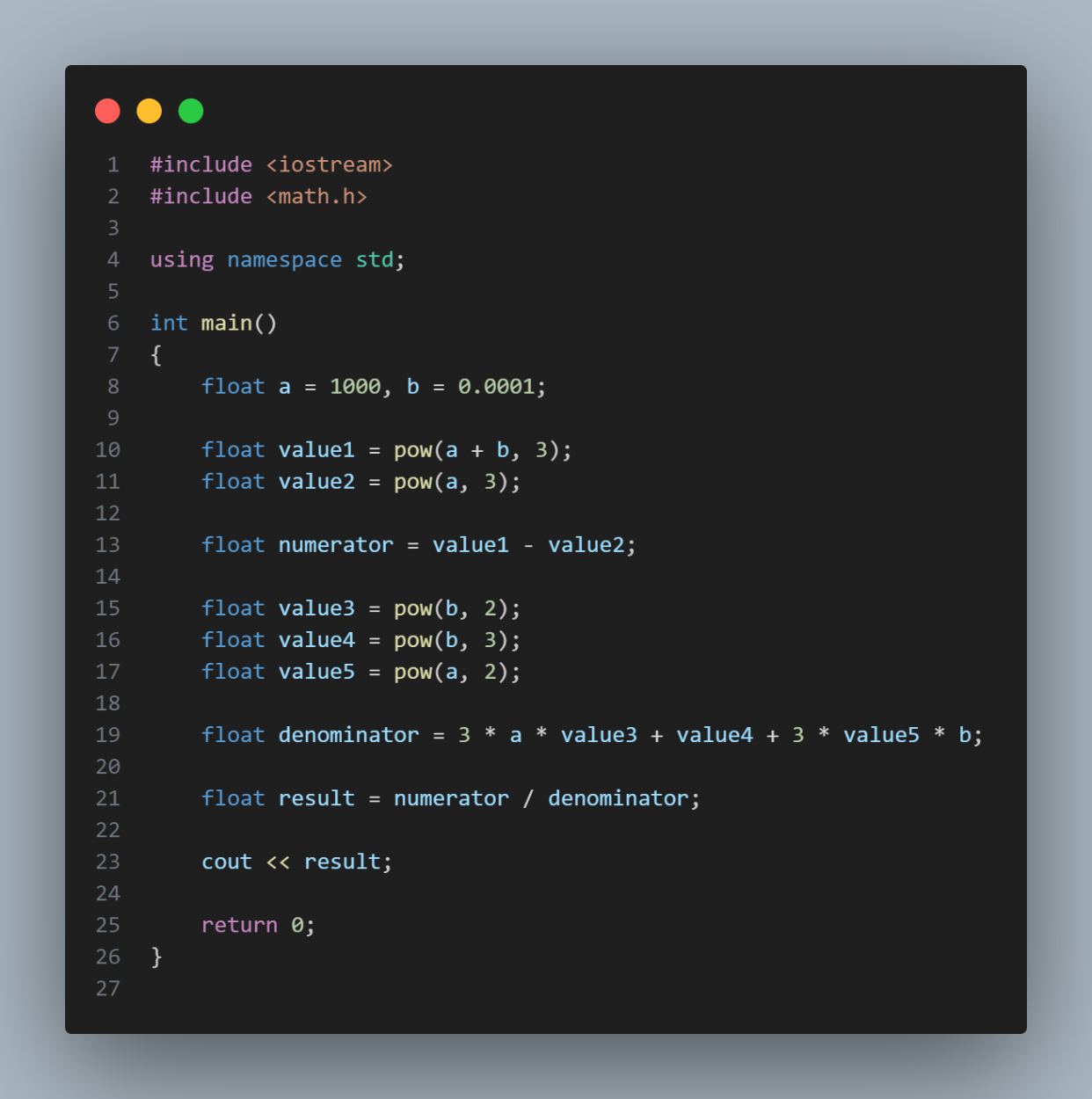


Рисунок 8a: Код до програми №1 (float)

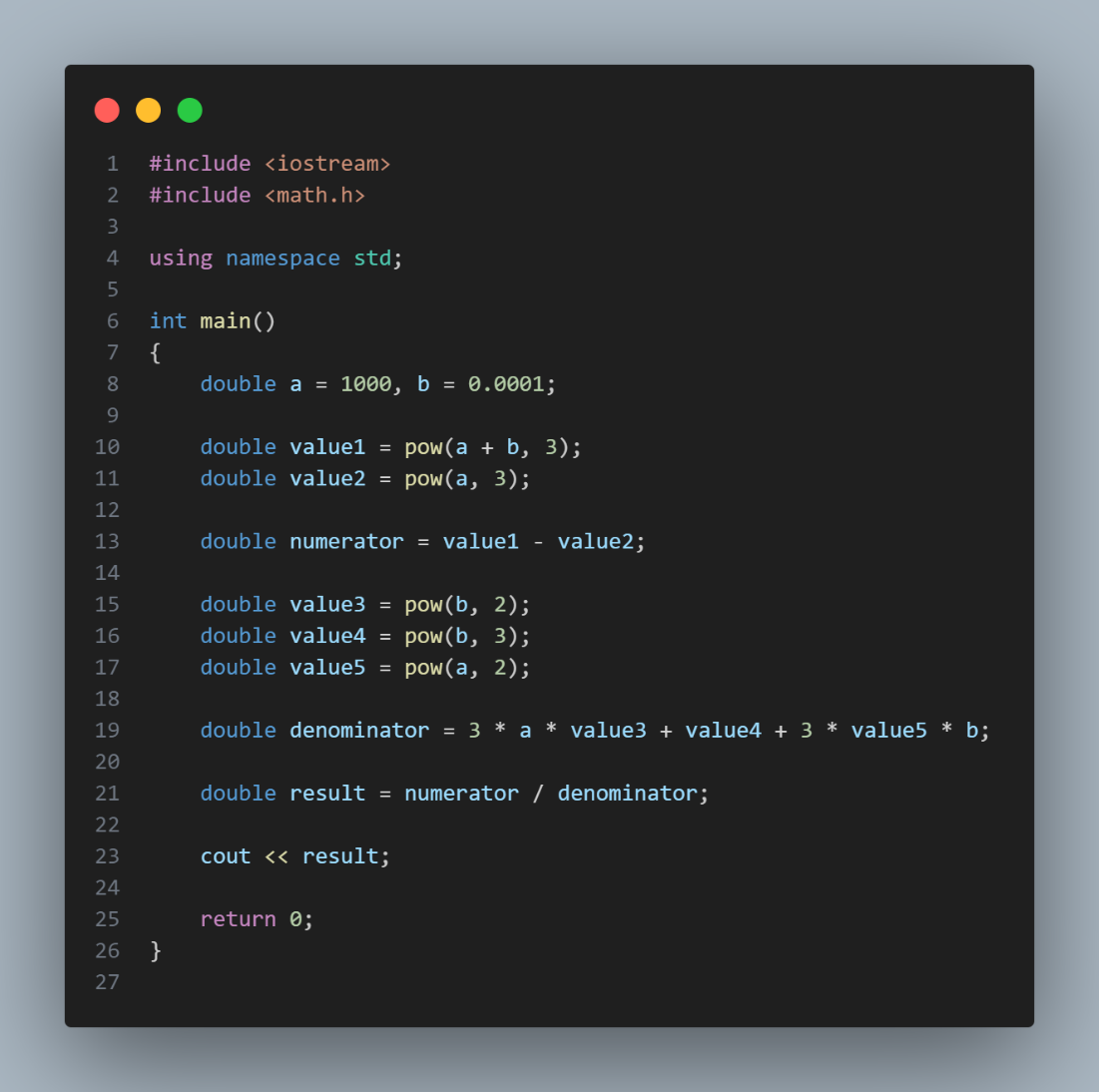


Рисунок 9б: Код до програми №1 (double)

Спочатку програма обчислює проміжні значення для заданих а і б, потім вона обчислює окремо значення для чисельника і знаменника. Після цього знаходить частку і виводить її в консоль.

https://github.com/artificial-intelligence-department/ai\_programming\_playground/pull/119

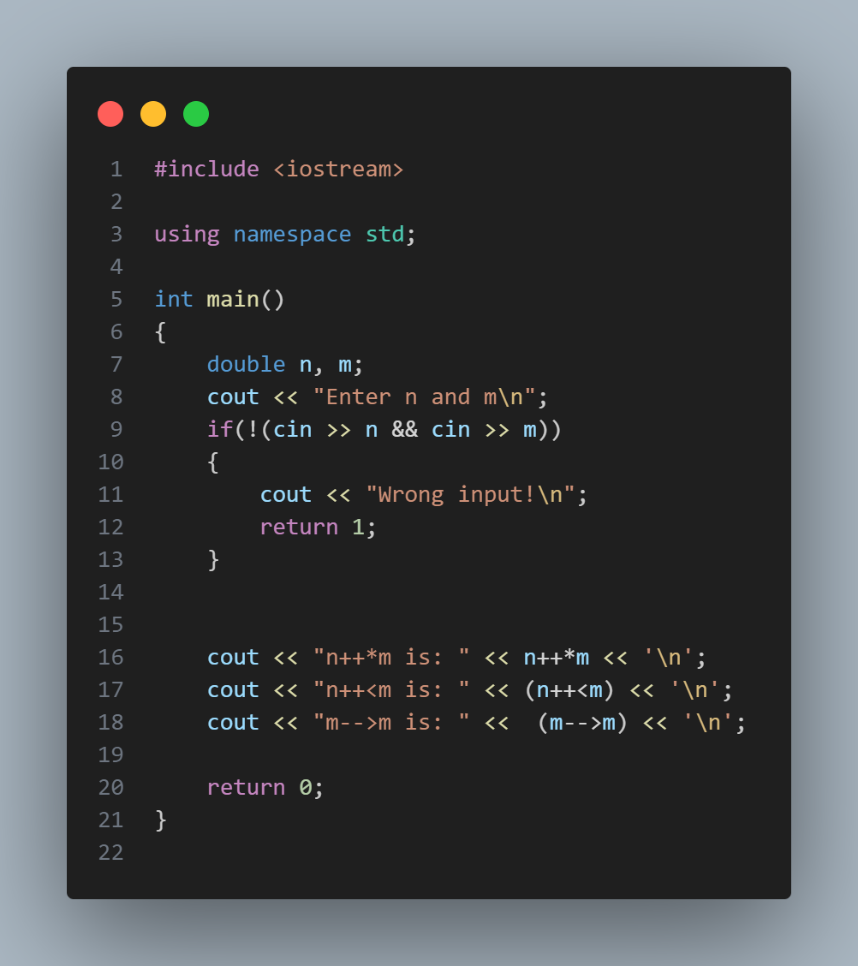
Завдання №2 VNS Lab 1 - Task 1-2  


Рисунок 10: Код до програми №2

Програма отримує значення н і м з консолі. Перевіряє чи дані передані у числовому форматі, далі виводить у консоль відповідні значення для трьох випадків.

https://github.com/artificial-intelligence-department/ai\_programming\_playground/pull/119

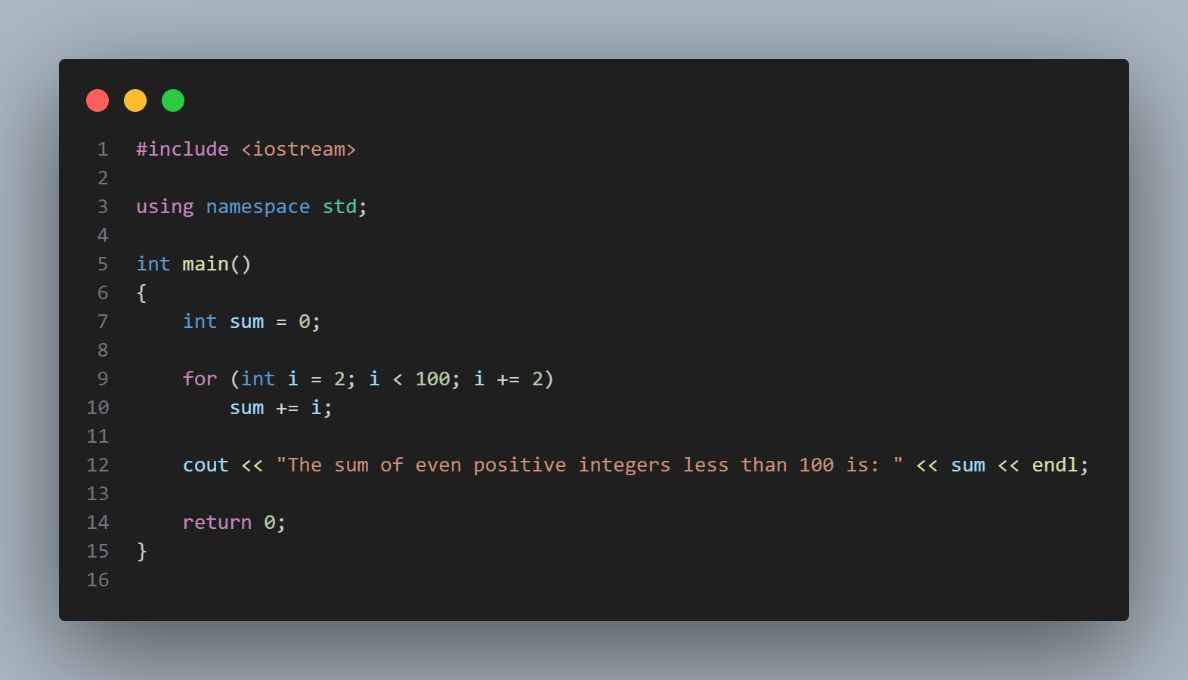
Завдання №3 VNS Lab 2  


Рисунок 11: Код до програми №3

Програма знаходить суму цілих додатних парних чисел менших 100, використовуючи цикл for. Отриманий результат програма виводить у консоль.

https://github.com/artificial-intelligence-department/ai\_programming\_playground/pull/119

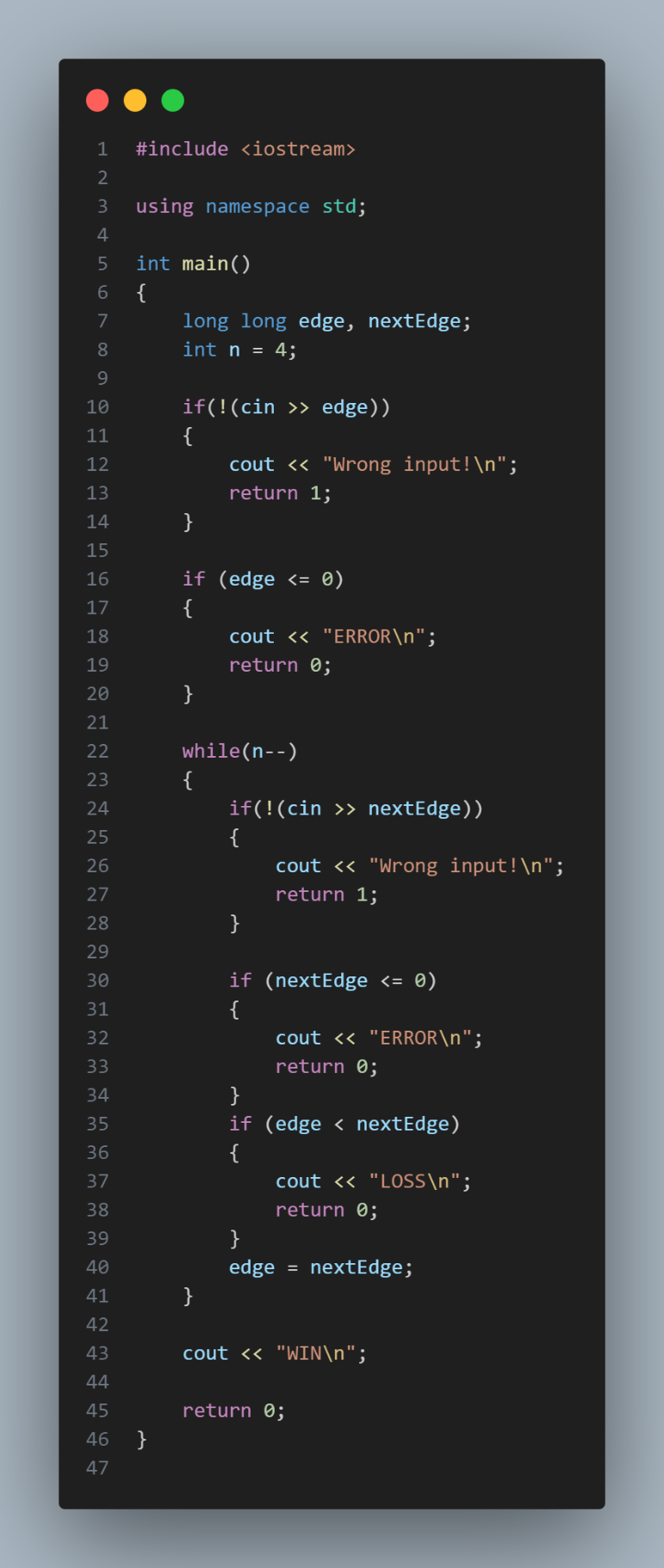
Завдання №4 Algotester Lab 1  


Рисунок 12: Код до програми №4

Спочатку програма отримує перше значення та одразу перевіряє чи воно додатне. Далі вона по черзі отримує ще 4 значення, перевіряє їх на додатність та порівнює їх із попередніми значеннями. Потім програма виводить відповідний результат у консоль.

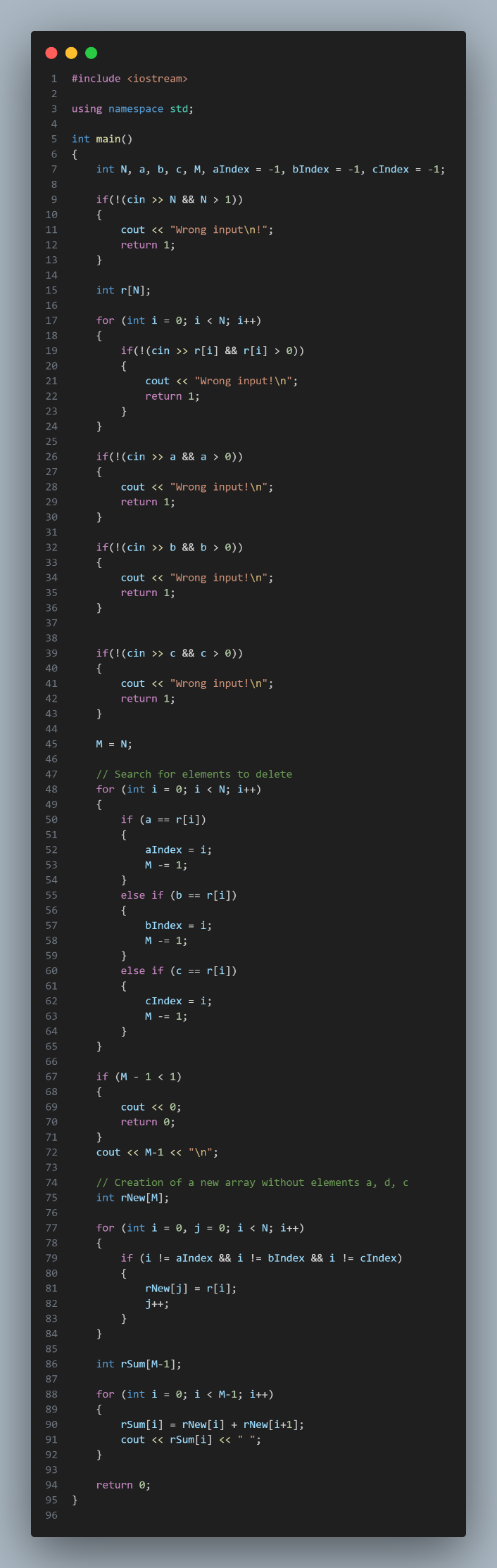
https://github.com/artificial-intelligence-department/ai\_programming\_playground/pull/119  
  
  
Завдання №5 Algotester Lab 2  


Рисунок 13: Код до програми №5

Програма отримує значення довжини масиву та створює його. Далі вона записує наступні отримані значення у цей масив. Також, вона записує 3 наступні отриманні значення. Далі перевіряє, значення для видалення дійсно належать масиву та запам’ятовує їх індекси. Програма створює новий масив без відповідних значень. Далі створює ще один масив довжиною на один менше за попередній, де кожен елемент - це сума сусідніх елементів попереднього масиву. Програма виводить довжину і елементи останнього масиву.

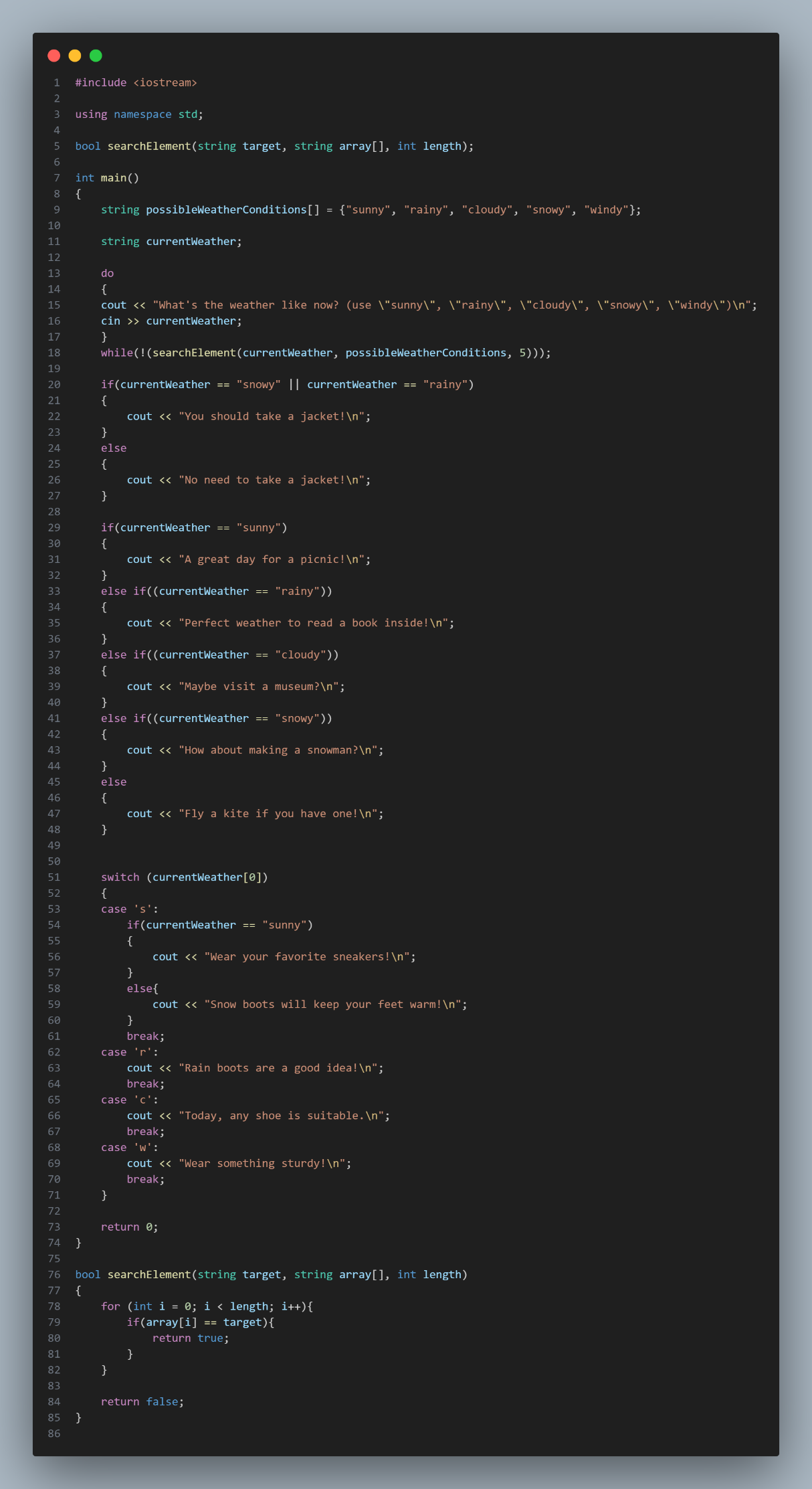
https://github.com/artificial-intelligence-department/ai\_programming\_playground/pull/119  
  
Завдання №6 Class Practice Work  


Рисунок 14: Код до програми №6

Програма отримує з консолі стан поточної погоди за перевіряє на відповідність введені дані користувача. Далі програма виводить пораду щодо верхнього одягу та активності відповідно до стану погоди. А далі програма дає пораду щодо взуття також відповідно до стану погоди.

https://github.com/artificial-intelligence-department/ai\_programming\_playground/pull/119  
  
Завдання №7 Self Practice Work  


Рисунок 15: Код до програми №7

Спочатку програма отримує довжину і ширину поля, а також номер кроку, координату якого треба знайти і перевіряє чи можливо зробити стільки кроків. Далі поступово знаходить координати кожного кроку, які відбувається вправо, вниз, вліво та вверх. Коли номер поточного кроку збігається із заданим, програма виводить у консоль шукані координати.

https://github.com/artificial-intelligence-department/ai\_programming\_playground/pull/119

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

Завдання №1  
Коли використовується тип даних float:

Рисунок 16а: Приклад виконання програми №1  
Коли використовується тип даних double:   


Рисунок 16б: Приклад виконання програми №1  
Час затрачений на виконання завдання 1 день.

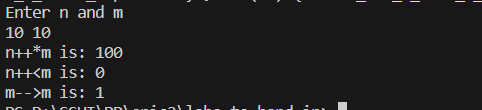
Завдання №2  


Рисунок 17: Приклад виконання програми №2

Час затрачений на виконання завдання 1 день.

Завдання №3  


Рисунок 18: Приклад виконання програми №3  
Час затрачений на виконання завдання 1 день.

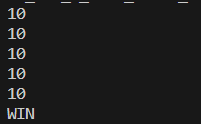
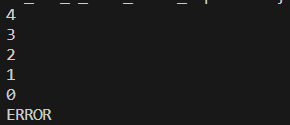
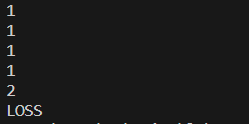
Завдання №4   
    
 

Рисунок 19: Приклади виконання програми №4

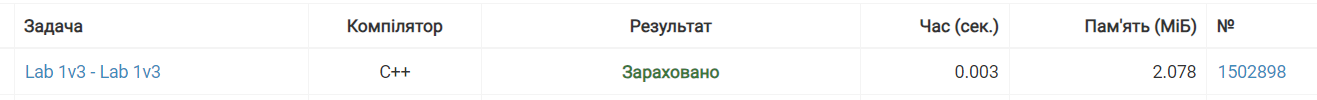


Рисунок 20: Завдання №4 зараховане в системі Алготестер

Час затрачений на виконання завдання 1 день.

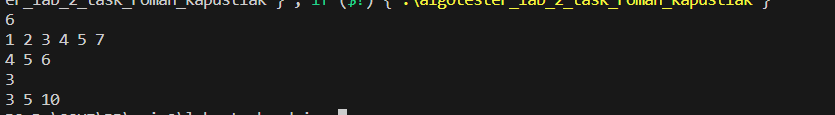
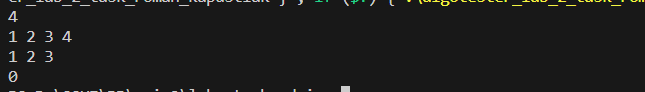
Завдання №5   
  


Рисунок 21: Приклади виконання програми №5



Рисунок 22: Завдання №5 зараховане в системі Алготестер

Час затрачений на виконання завдання 1 день.

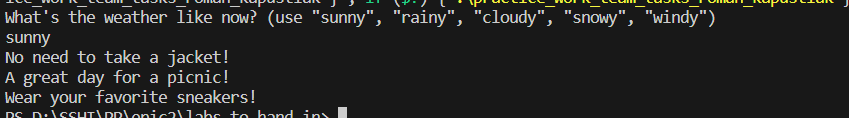
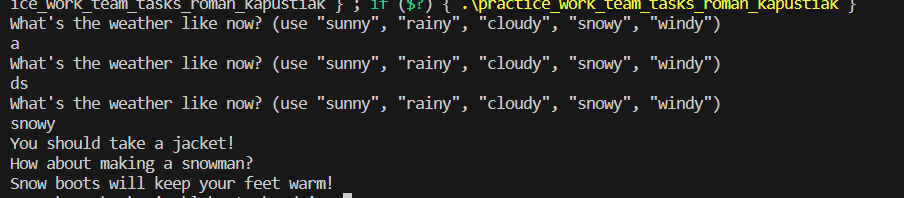
Завдання №6  
 

Рисунок 23: Приклади виконання програми №6

Час затрачений на виконання завдання 1 день.

Завдання №7  
  


Рисунок 24: Приклади виконання програми №7

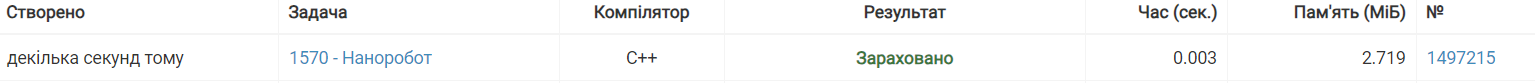


Рисунок 25: Завдання №7 зараховане в системі Алготестер

Час затрачений на виконання завдання 1 день.

# **Висновки:**

Під час роботи над цим розділом я ознайомився із принципами програмування на мові С++, а також практично закріпив отриманні знання за рахунок створення кількох програм із простою структурою лінійних і розгалужених алгоритмів, використовуючи основні оператори мови С++.

Пояснення результатів роботи програм із завдання 1:  
Через те, що у C++ десяткові вирази представляються як сума дробових значень, виникає проблема точності операцій з ними. Оскільки тип даних float використовує 4 байти, а double 8 байтів пам’яті, виникає певна різниця у точності їх застосування. Під час порівняння результатів, виникає похибка у 0,28, що була спричинена використанням даних типу float та проміжних змінних.

Пояснення результатів роботи програм із завдання 2:  
Дана поведінка програми відображає особливості роботи оператора інкременту та декременту. Спочатку відбувається зчитування поточного значення змінної і застосування його для поточних обчислень, а результат операції відповідно додавання, чи віднімання одиниці відображається лише у наступних обчисленнях.