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-13

Ласкавий Артем Олександрович

# **Тема роботи:**

1. Типи даних, змінні, константи
2. Ввід та вивід у С++
3. Операції
4. Умовні оператори

# **Мета роботи:**

1. Ознайомитись з основами синтаксису С++
2. Ознайомитись з поняттям змінних та констант
3. Ознайомитись з типами даних в С++ та їх відмінностями
4. Ознайомитись з видами операцій (унарні, порівняльні, логічні, операції присвоєння, бінарні, порозрядні)
5. Ознайомитись з умовними операторами (if else, if else if, switch case)
6. Ознайомитись з способами вводу та виводу у С++

# **Теоретичні відомості:**

1. Теоретичні відомості з переліком важливих тем:

Тема №1: Типи даних, змінні, константи

Тема №2: Ввід та вивід у С++

Тема №3: Операції

Тема №4: Умовні оператори

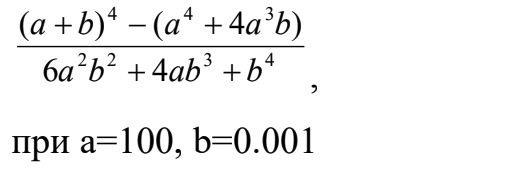
1. Індивідуальний план опрацювання теорії:

* Тема №1: Типи даних, змінні, константи
  + Джерела Інформації
    - Стаття.
      * <https://www.w3schools.com/cpp/cpp_variables.asp>
      * <https://www.w3schools.com/cpp/cpp_data_types.asp>
    - Відео.
      * [https://youtu.be/8jLOx1hD3\_o?si=kCxbvM3shI7qFzuK](file:////Users/user/Downloads/●https:/youtu.be/8jLOx1hD3_o%3fsi=kCxbvM3shI7qFzuK)
  + Що опрацьовано:
    - Опрацьовано основи синтаксису С++ (типи даних, змінні, константи).
  + Статус: Ознайомлений
  + Початок опрацювання теми: 20.08.2023
  + Звершення опрацювання теми: 26.09.2023
* Тема №2: Ввід та вивід у С++
  + Джерела Інформації:
    - Стаття.
      * <https://www.w3schools.com/cpp/cpp_output.asp>
    - Відео.
      * [https://youtu.be/8jLOx1hD3\_o?si=kCxbvM3shI7qFzuK](file:////Users/user/Downloads/●https:/youtu.be/8jLOx1hD3_o%3fsi=kCxbvM3shI7qFzuK)
  + Що опрацьовано:
    - Опрацьовані види вводу/виводу в С++, різницю між cin I scanf(), cout I printf()
  + Статус: Ознайомлений
  + Початок опрацювання теми: 20.08.2023
  + Звершення опрацювання теми: 30.09.2023
* Тема №3: Операції
  + Джерела Інформації:
    - Відео.
      * [https://youtu.be/8jLOx1hD3\_o?si=kCxbvM3shI7qFzuK](file:////Users/user/Downloads/●https:/youtu.be/8jLOx1hD3_o%3fsi=kCxbvM3shI7qFzuK)
    - Стаття.
      * <https://www.w3schools.com/cpp/cpp_operators.asp>
  + Що опрацьовано.
    - Опрацьовано види операції (унарні, порівняльні, логічні, операції присвоєння, бінарні, порозрядні)
  + Статус: Ознайомлений
  + Початок опрацювання теми: 20.08.2023
  + Звершення опрацювання теми: 01.10.2023
* Тема №4 Умовні оператори
  + Джерела Інформації:
    - Відео.
      * [https://youtu.be/8jLOx1hD3\_o?si=kCxbvM3shI7qFzuK](file:////Users/user/Downloads/●https:/youtu.be/8jLOx1hD3_o%3fsi=kCxbvM3shI7qFzuK)
    - Стаття.
      * <https://www.w3schools.com/cpp/cpp_conditions.asp>
  + Що опрацьовано:
    - Опрацьовані синтаксис та спосіб роботи операторів галуження (if else, if else if, switch case)
  + Статус: Ознайомлений
  + Початок опрацювання теми: 20.08.2023
  + Звершення опрацювання теми: 03.11.2023

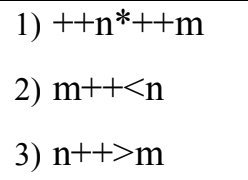
# **Виконання роботи:**

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

**Завдання №1 VNS Lab 1 - Task 1 Варіант №9**

* Деталі завдання: Обчислити значення виразу при різних дійсних типах даних (float й double). Обчислення варто виконувати з використанням проміжних змінних. Порівняти й пояснити отримані результати.
* 
* Важливі деталі для врахування в імплементації програми: Обчислення варто виконувати при різних дійсних типах даних (float й double), отже точність обрахунків та кінцевий результат може відрізнятися.

**Завдання №2 VNS Lab 1 - Task 2 Варіант №9**

* Деталі завдання: Обчислити значення виразів. Пояснити отримані результати.
* 
* Важливі деталі для врахування в імплементації програми: Префіксний інкремент збільшує змінну до використання в поточній операції, а постфіксний інкремент – після.

**Завдання №3 Algotester Lab 1 Варіант №2**

* Деталі завдання:

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.

Програма приймає вхідні дані – 4 цiлих числа h1,2,3,4 - довжини нiжок стола, 4 цiлих числа d1,2,3,4 - довжина, яку вiдпиляють вiд вiдповiдної ножки. Необхідно вивести вихідні дані – YES - якщо стiл буде стояти паралельно площинi пiдлоги та довжина найменшої нiжки не буде рiвна нулю.

ERROR - у випадку якщо ви вiдпиляєте бiльшу довжину нiж має ножка

NO - у iнших випадках

Обмеження:

0 ≤ h1,2,3,4 ≤ 1012

0 ≤ d1,2,3,4 ≤ 1012

* Важливі деталі для врахування в імплементації програми: для того, щоб програма працювала коректно при введенні будь-яких чисел в межах обмежень необхідно використати тип даних long int. Стіл може перевернутися i мiж початком та кiнцем вiдпилювання, наприклад коли вiдпиляють 2, але ще не встигнуть вiдпиляти 3тю нiжку. Навiть якщо стiл перевернеться – все одно можна отримати ERROR.

**Завдання № 4 Class Practice Work**

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

* Важливі деталі для врахування в імплементації програми: необхідно врахувати усі випадки вводу користувачем даних, в тому числі недійсні.

**Завдання №5 Self Practice Work – Офісна вулиця. Частина 1 (Алготестер)**

Деталі завдання: Зустрілися якось працівники великих компаній і почали... Обговорювати план вулиці. Виявляється, всі приміщення, які орендуватимуть ці компанії, збудують вздовж однієї вулиці. i-та компанія орендуватиме офіс довжиною li метрів. Офіси будуватимуть один за одним, починаючи з точки 0. Всі працівники приїжджатимуть на стоянку, яку побудують в точці 0, та будуть йти до офісів своїх компаній. Тобто, якщо офіси будуть збудовані в порядку p1,p2,...,pn, то перший офіс почнеться в точці 0 і закінчиться в точці lp1, другий почнеться в lp1 і закінчиться в lp1+lp2 і т.д. Двері кожного офісу завжди є в кінці будинку, який є ближчим до стоянки. Ваше завдання — допомогти розмістити офіси компаній на цій вулиці в такому порядку, щоб сумарна відстань від точки 0 до усіх офісів була мінімальною.

# Вхідні дані

У першому рядку задане ціле число n— кількість компаній.

У наступному рядку задано n цілих чисел li через пробіл — довжини офісів усіх компаній.

# Вихідні дані

У єдиному рядку виведіть n чисел від 1 до n — порядок компаній, в якому варто будувати офіси.

Якщо існує декілька оптимальних порядків — виведіть будь-який із них.

Обмеження

1 ≤ n ≤ 105 ,

1 ≤ li ≤ 104 .

* Важливі деталі для врахування в імплементації програми: Вихідні дані – порядок компаній, тому сортуючи вектори або масиви за довжиною, потрібно зберегти зв’язок між довжиною та індексом компанії. Це можна реалізувати структурою std::pair.

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

**Програма №1 VNS Lab 1 - Task 1 Варіант №9**

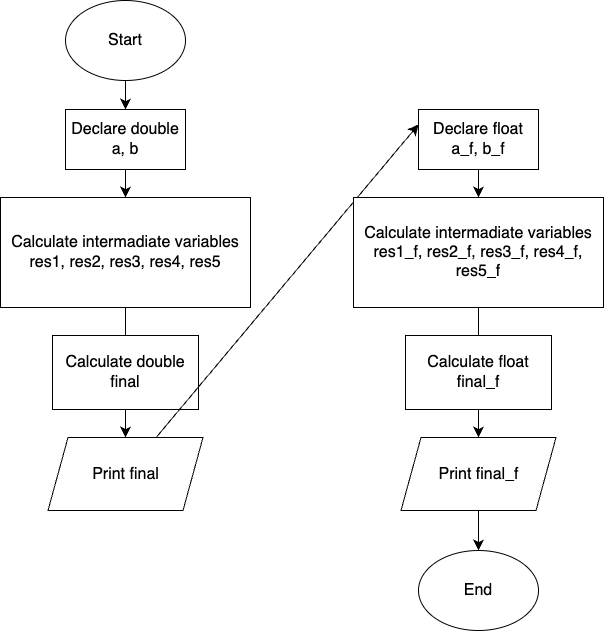
* Блок-схема:
* 

Figure 1 – Flowchart №1 VNS Lab 1 - Task 1 Варіант №9

* Планований час на реалізацію: 2 години

**Програма №2 VNS Lab 1 - Task 2 Варіант №9**

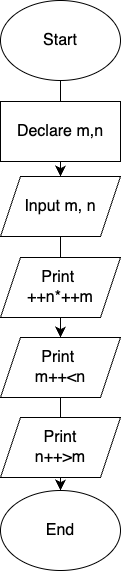
* Блок-схема:
* 

Figure 2 – Flowchart №2 VNS Lab 1 - Task 2 Варіант №9

* Планований час на реалізацію: 30 хвилин.

**Програма №3 Algotester Lab 1 Варіант №2**

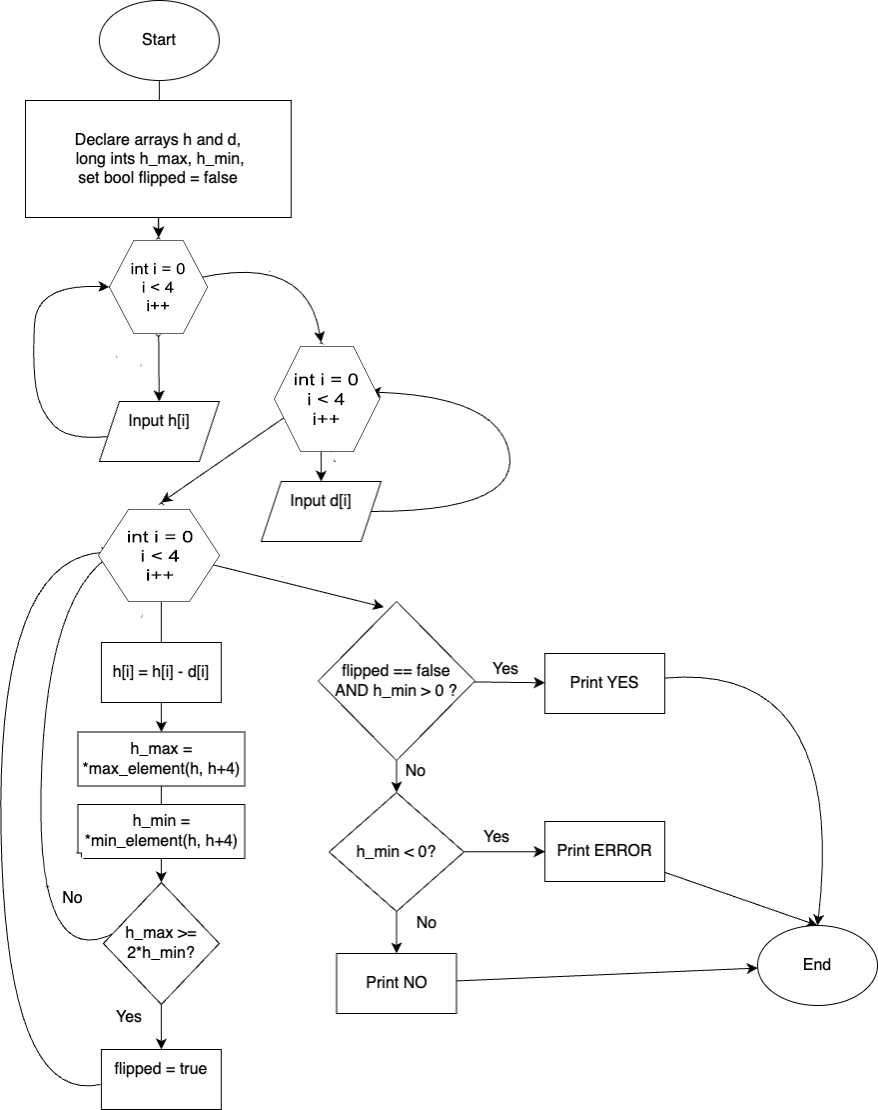
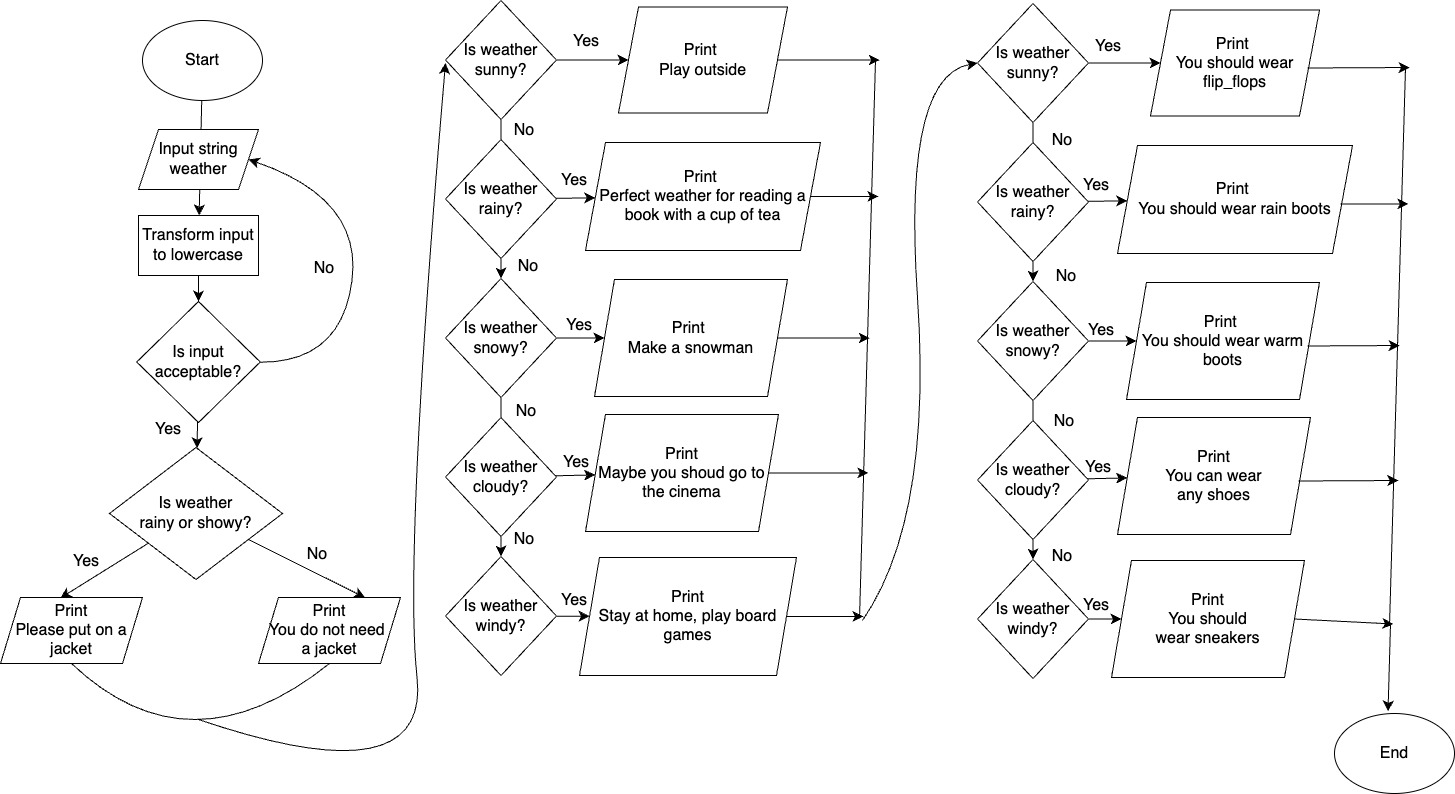
* Блок-схема:
* 

Figure 3 – Flowchart №3 Algotester Lab 1 Варіант №2

* Планований час на реалізацію: 2 години

**Програма № 4 Class Practice Work**

* Блок-схема:
* 
* Планований час на реалізацію: 2 години

**Програма №5 Self Practice Work – Офісна вулиця. Частина 1 (Алготестер)**

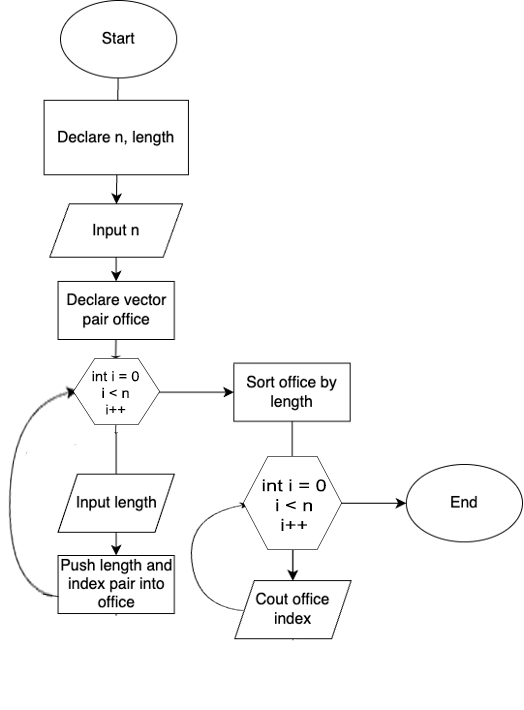
* Блок-схема:
* 

Figure 4 – Flowchart №5 Self Practice Work – Офісна вулиця. Частина 1 (Алготестер)

* Планований час на реалізацію: 2 години

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

Без змін.

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

Завдання №1 Деталі по програмі + Вставка з кодом з підписами до вставки. Посилання на файл програми у пул-запиті GitHub

**Завдання №1 VNS Lab 1 - Task 1 Варіант №9**

Посилання на файл програми у пул-запиті: <https://github.com/artificial-intelligence-department/ai_programming_playground/blob/epic_2_pactice_and_labs_artem_laskavyi/ai_13/artem_laskavyi/epic%202/vns_lab_1_task_1_artem_laskavyi.cpp>

#include <iostream>

#include <cmath>

using namespace std;

int main()

{

double a = 100.0;

double b = 0.0001;

double res1 = pow(a+b, 4);

double res2 = (pow(a, 4)+4\*pow(a, 3)\*b);

double res3 = (6\*pow(a, 2)\*pow(b, 2));

double res4 = 4\*a\*pow(b, 3);

double res5 = pow(b, 4);

double final = (res1 - res2)/(res3 + res4+ res5);

cout << final << endl;

cout << "--------------------" << endl;

float a\_f = 100;

float b\_f = 0.0001;

float res1\_f = pow(a\_f+b\_f, 4);

float res2\_f = (pow(a\_f, 4)+4\*pow(a\_f, 3)\*b\_f);

float res3\_f = (6\*pow(a\_f, 2)\*pow(b\_f, 2));

float res4\_f = 4\*a\_f\*pow(b\_f, 3);

float res5\_f = pow(b\_f, 4);

float final\_f = (res1\_f - res2\_f)/(res3\_f +res4\_f+res5\_f);

cout << final\_f << endl;

return 0;

}

Figure 5 – Завдання №1 VNS Lab 1 - Task 1 Варіант №9

**Завдання №2 VNS Lab 1 - Task 2 Варіант №9**

Посилання на файл програми у пул-запиті: <https://github.com/artificial-intelligence-department/ai_programming_playground/blob/epic_2_pactice_and_labs_artem_laskavyi/ai_13/artem_laskavyi/epic%202/vns_lab_1_task_2_artem_laskavyi.cpp>

#include <iostream>

using namespace std;

int main()

{

double m,n;

cin >> m >> n;

cout << ++n\*++m<< endl;

cout << (m++<n) << endl;

cout << (n++>m) << endl;

return 0;

}

Figure 6 – Завдання №2 VNS Lab 1 - Task 2 Варіант №9

**Завдання №3 Algotester Lab 1 Варіант №2**

Посилання на розв’язок в Алготестері: <https://algotester.com/en/ProblemSolution/Display/1502186>

Посилання на файл програми у пул-запиті: <https://github.com/artificial-intelligence-department/ai_programming_playground/blob/epic_2_pactice_and_labs_artem_laskavyi/ai_13/artem_laskavyi/epic%202/algotester_lab_1_artem_laskavyi.cpp>

#include <iostream>

using namespace std;

int main(){

long int h[4]; //довжина ніжок

long int d[4]; //довжина відпилу

bool flipped = false;

long int h\_max, h\_min;

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

cin >> h[i];

}

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

cin >> d[i];

}

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

h[i] = h[i] - d[i];

//min\_element and max\_element return an iterator i.e a pointer

// so dereferencning is needed

h\_max = \*max\_element(h, h+4);

h\_min = \*min\_element(h, h+4);

if(h\_max >= 2\*h\_min){

flipped = true;

}

}

if(flipped == false && h\_min > 0){

cout << "YES" << endl;

}else if(h\_min < 0){

cout << "ERROR" << endl;

}else{

cout << "NO" << endl;

}

return 0;

}

Figure 7 – №3 Algotester Lab 1 Варіант №2

**Завдання № 4 Class Practice Work**

Посилання на файл програми у пул-запиті: <https://github.com/artificial-intelligence-department/ai_programming_playground/blob/epic_2_pactice_and_labs_artem_laskavyi/ai_13/artem_laskavyi/epic%202/practice_work_team_tasks_artem_laskavyi.cpp>

#include <iostream>

#include <string>

using namespace std;

int main(){

string weather;

weatherInput:

cin >> weather;

transform(weather.begin(), weather.end(), weather.begin(), ::tolower);

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

cout << "My advice for " << weather << " weather is:\n";

}else{

cout << "Invalid input. Try again\n";

goto weatherInput;

}

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

cout << "Please put on a jacket!\n";

}else{

cout << "You do not need a jacket\n";

}

if (weather == "sunny"){

cout << "Play outside\n";

}else if(weather == "rainy"){

cout << "Perfect weather for reading a book with a cup of tea\n";

}else if(weather == "snowy"){

cout << "Make a snowman!\n";

}else if(weather == "cloudy"){

cout << "Maybe you should go to the cinema?\n";

}else if(weather == "windy"){

cout << "Stay at home, play board games\n";

}

char sunny = 's', cloudy = 'c', rainy = 'r', snowy = 's', windy = 'w';

switch (weather[0]){

case 's':

if(sunny){

cout << "You should wear flip-flops\n";

break;

}else if(snowy){

cout << "You should wear warm boots\n";

break;

}

case 'c':

cout << "You can wear any shoes\n";

break;

case 'r':

cout << "You should wear rain boots\n";

break;

case 'w':

cout << "You should wear sneakers\n";

break;

}

return 0;

}

Figure 8 – Завдання № 4 Class Practice Work

**Завдання №5 Self Practice Work – Офісна вулиця. Частина 1 (Алготестер)**

Посилання на розв’язок в Алготестері: <https://algotester.com/en/ProblemSolution/Display/1503227>

Посилання на файл програми у пул-запиті: <https://github.com/artificial-intelligence-department/ai_programming_playground/blob/epic_2_pactice_and_labs_artem_laskavyi/ai_13/artem_laskavyi/epic%202/practice_work_self_algotester_tasks_artem_laskavyi.cpp>

#include <iostream>

#include <algorithm>

#include <vector>

using namespace std;

int main() {

unsigned int n, length;

cin >> n;

vector<pair<int,int >> office;

for (int i=0; i<n; i++){

cin >> length;

office.push\_back({length, i+1});

}

sort(office.begin(), office.end());

for (int i=0; i<n; i++){

cout << office[i].second << endl;

}

return 0;

}

Figure 9 – Завдання №5 Self Practice Work – Офісна вулиця. Частина 1 (Алготестер)

Пул-запит: <https://github.com/artificial-intelligence-department/ai_programming_playground/pull/259>

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

**Завдання №1 VNS Lab 1 - Task 1 Варіант №9**

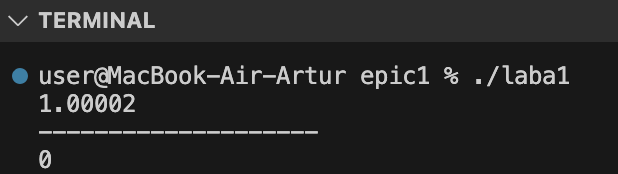


Figure 10 – VNS Lab 1 Test1

Порівняння значень змінних типу double та float в режимі Run and Debug:

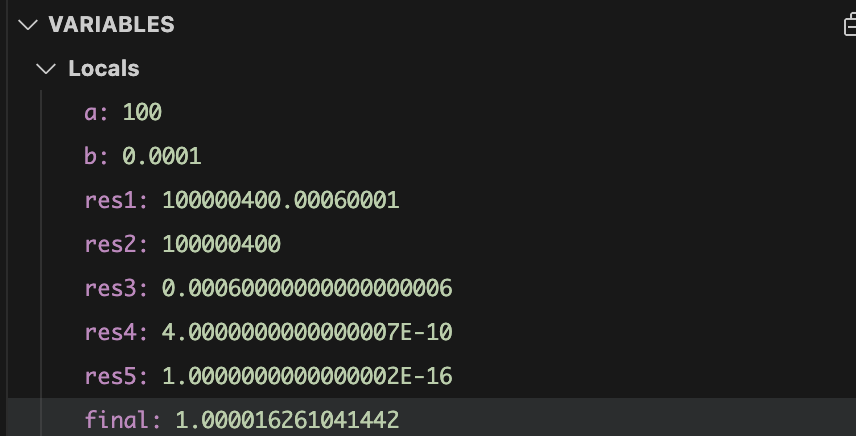


Figure 11 – VNS Lab 1 Test2

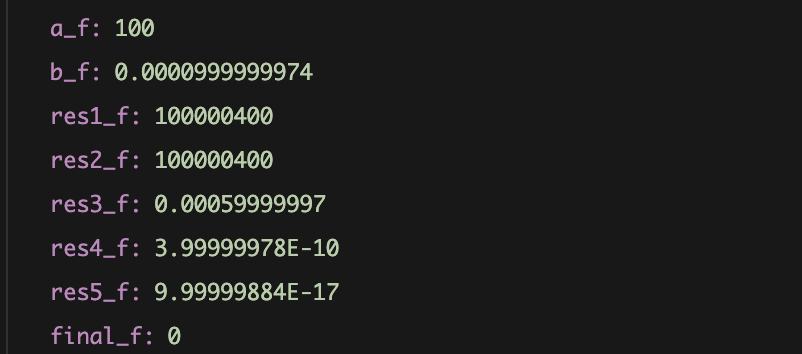


Figure 12 – VNS Lab 1 Test3

Час затрачений на виконання завдання: 1 година.

**Завдання №2 VNS Lab 1 - Task 2 Варіант №9**

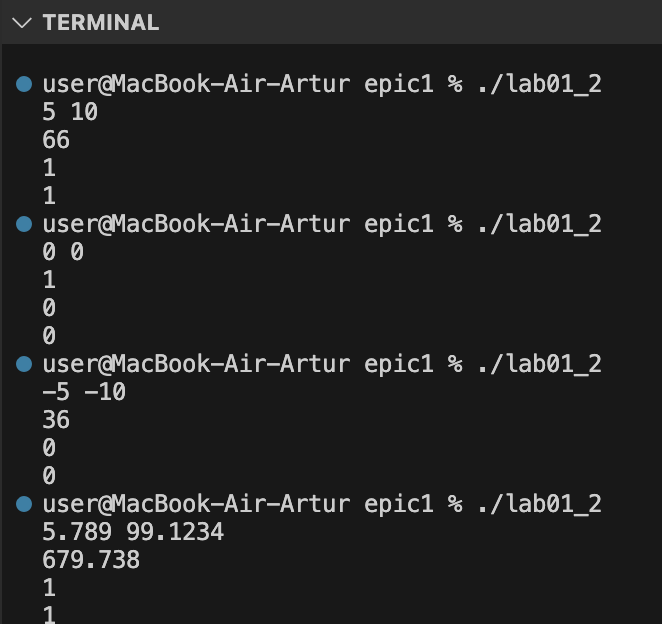


Figure 13 – VNS Lab 1 Task 2 Test1

Час затрачений на виконання завдання: 10 хвилин.

**Завдання №3 Algotester Lab 1 Варіант №2**

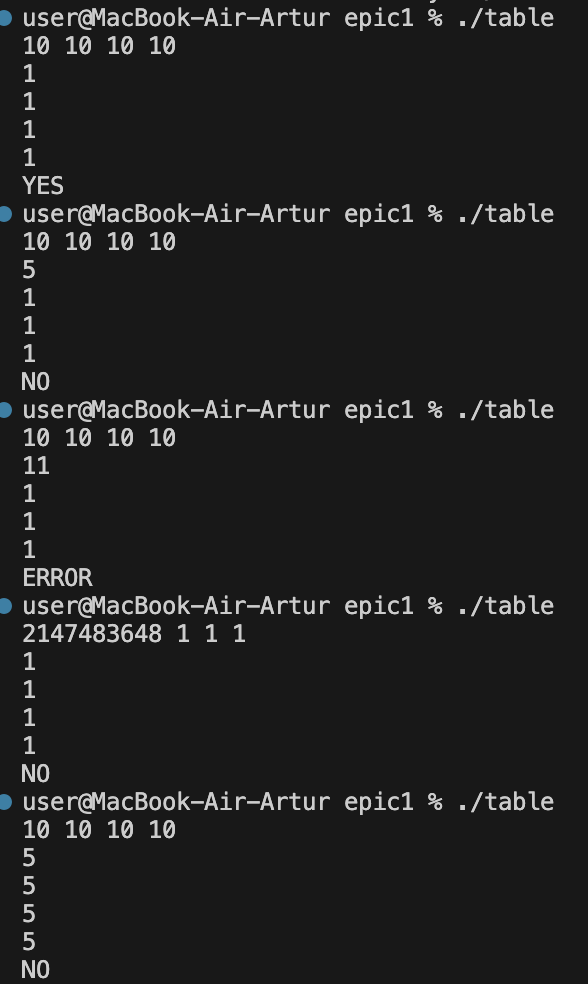


Figure 14 – Algotester Lab 1 Варіант №2 Test1

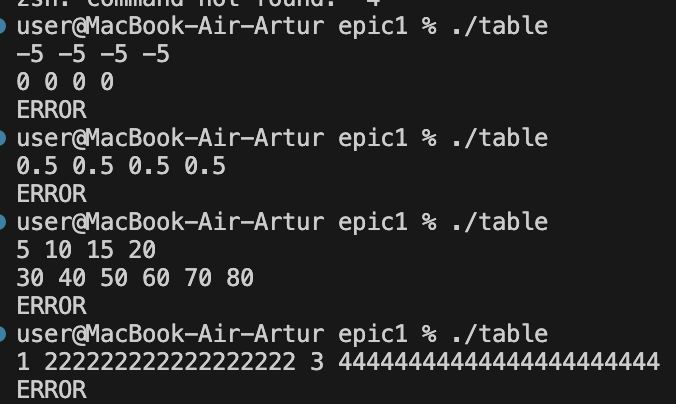


Figure 15 – Algotester Lab 1 Варіант №2 Test2

Час затрачений на виконання завдання: 3 години.

**Завдання № 4 Class Practice Work**

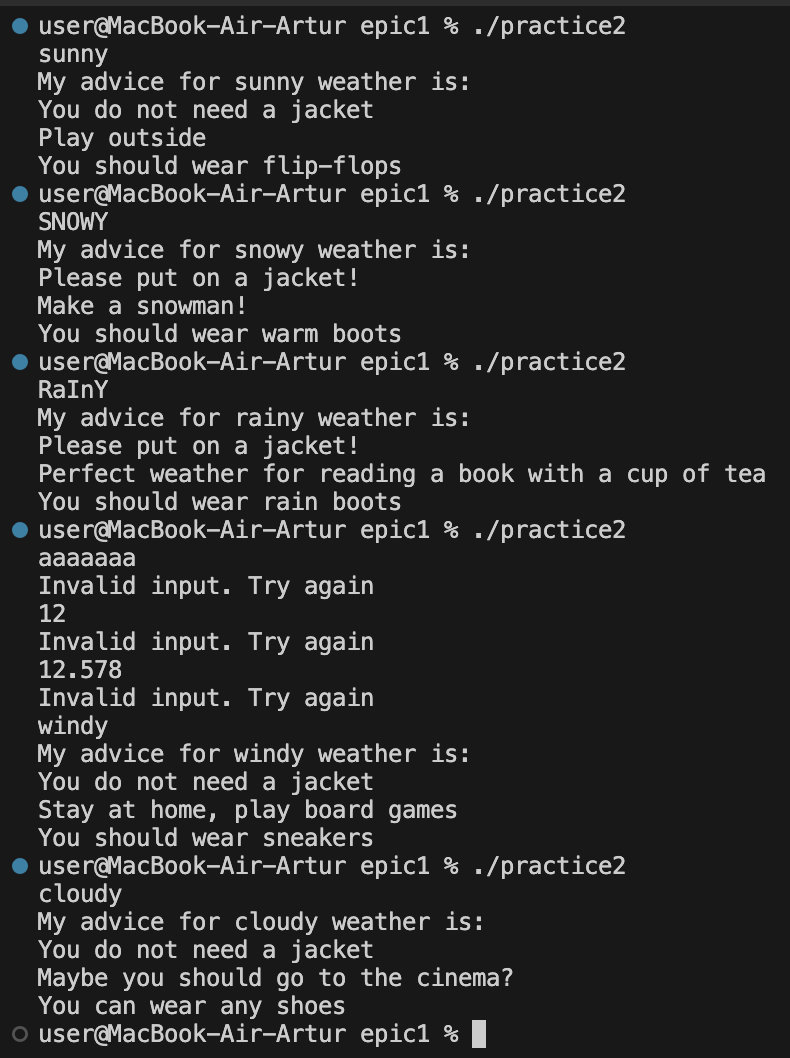


Figure 16 – Class Practice Work Test

Час затрачений на виконання завдання: 1 година.

**Завдання №5 Self Practice Work – Офісна вулиця. Частина 1**

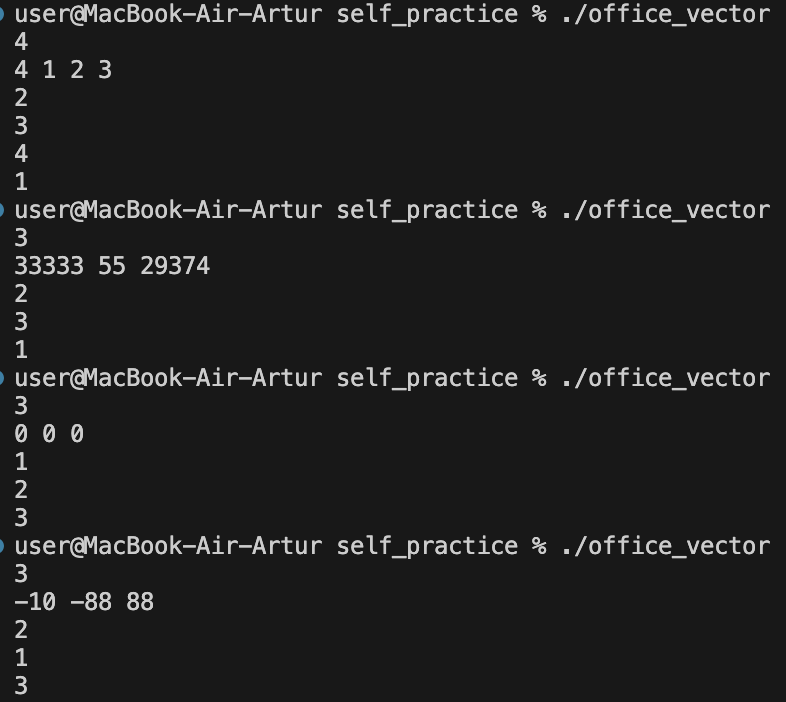
****

Figure 17 – Self Practice Work – Офісна вулиця. Частина 1 Test

Час затрачений на виконання завдання: 4 години.

# **Висновки:**

Під час виконання епіку 2 я ознайомився з основами синтаксису С++, змінних та констант, типами даних, видами операцій, умовними операторами, способами вводу та виводу. На практиці було реалізовано програми з використанням типів даних float та double, постфіксного та префіксного інкрементів, умовних операторів if else, if else if, switch case, cin, cout/