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

**Виконала:**

Студентка групи ШІ-13

Жмуд Анастасія Анатоліївна

# **Тема роботи:**

Використання основних операторів мови С для лінійних та розгалужених алгоритмів, алгоритмів циклу. Типи даних та оголошення змінних. Умовні оператори.

# **Мета роботи:**

Набуття знань про різноманітні типи даних, їх використання. Отримання навичок у роботі з операторами мов С та С++. Знайомство алгоритмами розгалуження й циклу. Ітераційні процеси.

# **Теоретичні відомості:**

1. Теоретичні відомості з переліком важливих тем:

* **Тема №1**: Типи даних.
* **Тема №2**: Алгоритми циклу.
* **Тема №3**: Алгоритми розгалуження.

1. Індивідуальний план опрацювання теорії:

* **Тема №1**: Типи даних
  + Джерела Інформації
    - <http://cpp.dp.ua/typy-danyh/>
    - <https://w3schoolsua.github.io/c/c_data_types.html#gsc.tab=0>
  + Що опрацьовано:
    - Інформація про різні типи даних та їх обсяг
    - Приклади оголошення змінних для різних типів даних
  + Статус: Ознайомлений
  + Початок опрацювання теми: 12.10.2023
  + Звершення опрацювання теми: 13.10.2023
* **Тема №2**: Алгоритми циклу
  + Джерела Інформації:
    - <https://step.org.ua/konspekt/algoritm/tema5>
    - <https://cherto4ka.xyz/2018/10/21/%D1%86%D0%B8%D0%BA%D0%BB%D0%B8-%D0%B2-%D1%81/>
    - <https://acode.com.ua/urok-72-tsykl-for/>
  + Що опрацьовано:
    - Матеріал про види операторів циклу для С та С++, їх принцип роботи
    - Локальні та глобальні змінні
    - Цикл з лічильником
  + Статус: Ознайомлений
  + Початок опрацювання теми: 15.10.2023
  + Звершення опрацювання теми: 20.10.2023
* **Тема №3**: Алгоритми розгалуження
  + Джерела Інформації:
    - <http://nikolay.in.ua/distantsijne-navchannya/6-klas/870-algoritmi-z-rozgaluzhennyami>
    - <https://acode.com.ua/urok-67-operatory-umovnogo-rozgaluzhennya-if-else/>
    - <https://www.slideshare.net/AlexanderKuzmenko/c-69563261>
  + Що опрацьовано:
    - Основна інформація про алгоритми розгалуження
    - Оператори розгалуження в С та С++
    - Оператор вибору Switch
  + Статус: Ознайомлений
  + Початок опрацювання теми: 18.10.2023
  + Звершення опрацювання теми: 20.10.2023

# **Виконання роботи:**

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

**Завдання №1 Опрацювання теорії**

* Деталі завдання:

Опрацювання різноманітних матеріалів, аналіз та дослідження відео, статей, книг на задані теми. Організація та структурування отриманих даних для можливості ефективніше засвоювати отримані знання на практиці. Вивчення найважливіших моментів.

**Завдання №2 Перегляд вимог та проектування**

* Деталі завдання:

Введення завдань в Trello, перегляд задач, та створення математичної моделі. Проектування блок-схем в Draw.io.

* Важливі деталі:

Використання різноманітних фігур для певних дій та особливості проектування flow-charts, що включають в себе алгоритми циклу.

**Завдання №3 VNS Lab 1 Task 1**

* Деталі завдання:

Обчислити значення виразу при різних дійсних типах даних. Обрахунки з використанням проміжних змінних. Порівняння й пояснення отриманих результатів.

* Важливі деталі для врахування в імплементації програми:

Потрібно провести 2 тести: 1 для типу даних double, 2 для типу float. Треба використати допоміжні змінні для зберігання проміжних результатів.

**Завдання №4 VNS Lab 1 Task 2**

* Деталі завдання:

Обчислити значення виразів при введені різних значень змінних.

- Важливі деталі для врахування в імплементації програми:

Врахування пріоритетності зчитування знаків (інкриментів, декриментів, знаків порівняння).

**Завдання №5 Algotester Lab1**

- Деталі завдання:

Є стiл, у якого є 4 нiжки, довжини яких дано. Ви хочете зробити нiжки рiвної довжини, для цього ви вiдпиляєте d вiд кожної нiжки.

Вказати чи пiсля усiх манiпуляцiй стiл буде цiлий та паралельний пiдлозi. Якщо довжина, яку вiдрiжуть буде бiльша за довжину ножки - вам треба вивести «ERROR».

Якщо він буде не паралельний підлозі – виводимо «NO», в іншому випадку – «YES».

- Важливі деталі для врахування в імплементації:

Навiть якщо стiл перевернеться - ви все одно вiдпилюєте нiжки i можете отримати ERROR.

**Завдання №6 Self Practice Task**

- Деталі завдання:

Задача Algotester «Коля, Вася і Теніс». Коля і Вася грають у теніс. Знаючи, хто переміг кожної подачі — виведіть загальний рахунок по партіях в грі Коля-Вася. А якщо вони не дограли останню партію, то і її рахунок теж.

- Важливі деталі для врахування в імплементації:

Партія вважається виграною, коли один з гравців набере не менше одинадцяти очок з перевагою щонайменше у два очки.

**Завдання №7 Class Practice Work**

- Деталі завдання:

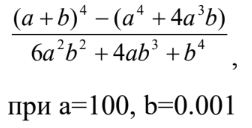
Створення простого порадника щодо погоди. Користувач вводить поточні погодні умови, а програма видає рекомендації щодо активності на основі погоди

- Важливі деталі для врахування в імплементації:

Врахування того, що ми не можемо судити в switch тільки по першій літері, адже у нас є 2 слова, що починаються на одну й ту ж літеру.

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

**Програма №1**

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

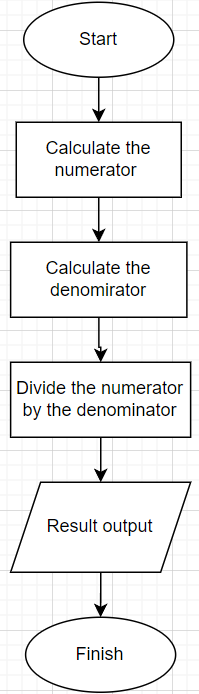
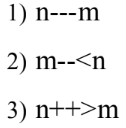
* Блок-схема:

Рисунок 1: Блок-схема до програми VNS lab1(Task 1)

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

Проводити обчислення для обох типів даних (float i double)

**Програма №2**

 Обчислити значення виразів.

* Блок-схема :

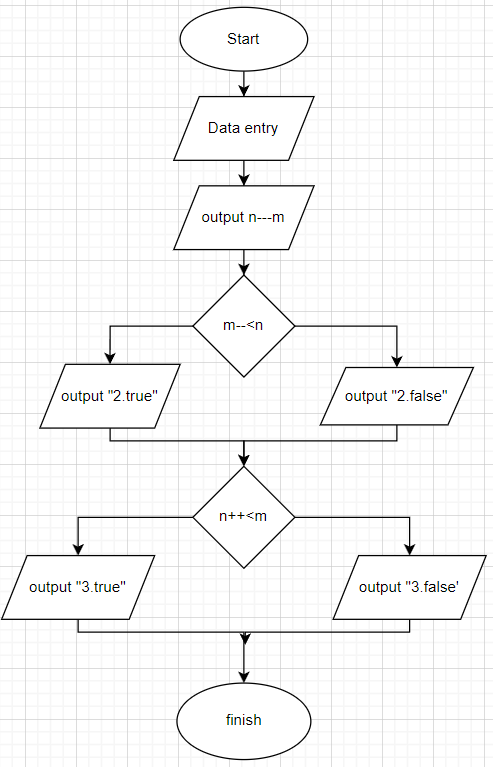


Рисунок 2: Блок-схема до програми VNS lab1(Task 2)

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

Врахування пріоритетності зчитування знаків (інкрементів, декриментів, знаків порівняння).

**Програма №3**

Є стiл, у якого є 4 нiжки, довжини яких дано. Ви хочете зробити нiжки рiвної довжини, для цього ви вiдпиляєте d вiд кожної нiжки.

Вказати чи пiсля усiх манiпуляцiй стiл буде цiлий та паралельний пiдлозi. Якщо довжина, яку вiдрiжуть буде бiльша за довжину ножки - вам треба вивести «ERROR».

Якщо він буде не паралельний підлозі – виводимо «NO», в іншому випадку – «YES.

* Блок-схема :

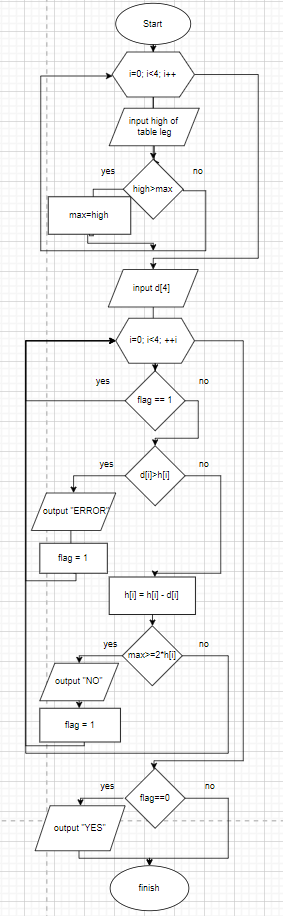


Рисунок 3: Блок-схема до програми Algotester lab1 (Task 2)

* Планований час на реалізацію: 3 години
* Важливі деталі для врахування в імплементації:

Навiть якщо стiл перевернеться - ви все одно вiдпилюєте нiжки i можете отримати

ERROR.

**Програма №5**

Коля і Вася грають у теніс. Знаючи, хто переміг кожної подачі — виведіть загальний рахунок по партіях в грі Коля-Вася. А якщо вони не дограли останню партію, то і її рахунок теж.

- Блок-схема:

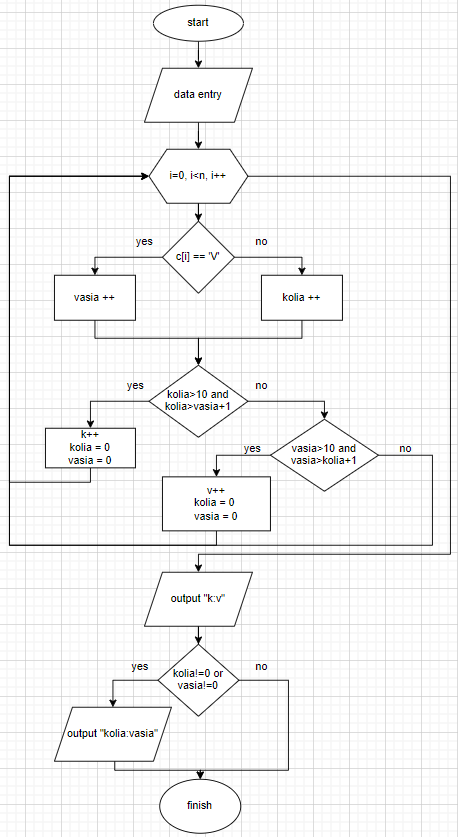


Рисунок 4: Блок-схема до програми Self Practice Task

- Планований час на реалізацію: 2 години

- Важливі деталі для врахування в імплементації програми:

Партія вважається виграною, коли один з гравців набере не менше одинадцяти очок з перевагою щонайменше у два очки.

**Програма №6**

Створення простого порадника щодо погоди. Користувач вводить поточні погодні умови, а програма видає рекомендації щодо активності на основі погоди.

-Блок-схема:

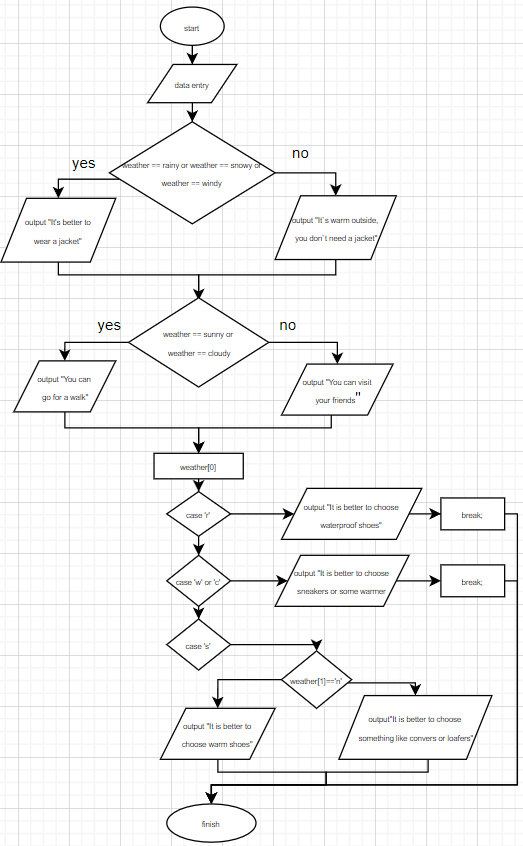


Рисунок 5: Блок-схема до програми Class Practice Work

- Планований час на реалізацію: 1 година

- Важливі деталі для врахування в імплементації програми:

Врахування того, що ми не можемо судити в switch тільки по першій літері, адже у нас є 2 слова, що починаються на одну й ту ж літеру.

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

**Завдання №1**

#include <stdio.h>

#include <math.h>

int main()

{

    double a = 1000, b = 0.0001, numerator, denominator, rez;

    numerator = pow(a+b,3)-(pow(a,3)+3\*1000\*1000\*0.0001);

    denominator = 3\*1000\*pow(b,2)+pow(b,3);

    rez = numerator/denominator;

    printf("%lf", rez);

}

#include <stdio.h>

#include <math.h>

int main()

{

    float a = 1000, b = 0.0001, numerator, denominator, rez;

    numerator = pow(a+b,3)-(pow(a,3)+3\*1000\*1000\*0.0001);

    denominator = 3\*1000\*pow(b,2)+pow(b,3);

    rez = numerator/denominator;

    printf("%f", rez);

}

Спочатку потрібно підключити усі необхідні для роботи програми бібліотеки, у цьому випадку це «stdio.h» та «math.h.» Потім відбувається оголошення змінних та присвоєння їм заданих значень (в першому коді ми оголосили їх типом double, а в другому – float. Після, по черзі обчислюємо чисельник і знаменник дробу за формулою даною в умові, ділимо чисельник на знаменник. Кінцевим етапом є форматований вивід результату.

**Завдання №2**

#include <stdio.h>

int main(){

    int m, n;

    printf("Enter m:");

    scanf("%d", &m);

    printf("Enter n:");

    scanf("%d", &n);

        printf("1.");

        printf("%d\n", n---m);

        if(m--<n){

         printf("2.true\n");

         }

        else{

         printf("2.false\n");

        }

        if(n++>m){

            printf("3.true");

        }

        else{

            printf("3.false");

        }

}

Першим етапом роботи для виконання даної задачі є підключення бібліотеки «stdio.h» Після, необхідно ввести усі змінні, у нашому випадку n та m. Далі проводимо обчислення першого виразу та виводимо результат. Наступним іде оператор if у якому ми задаємо умову і в залежності від того, чи справджується вона чи ні виводимо булеве значення true або false. Таку ж саму операцію проводимо для наступної умови і виводимо булеве значення.

**Завдання №3**

#include <iostream>

#include <string>

using namespace std;

int main(){

    long long h[4], max=0, flag=0, sum=0, sum1=0;

    long long d[4];

    string rez = "YES";

    for (int i=0; i<4; i++){

        cin>>h[i];

        if(h[i]>max){

            max=h[i];

        }

        sum+=h[i];

    }

    for ( int i=0; i<4; i++){

        cin>>d[i];

        sum1+=d[i];

    }

    if(sum+sum1==0||sum1==0){

        cout<<"YES";

        return 0;

    }

    for (int i=0; i<4; i++){

        if(d[i]>h[i]){

            cout<<"ERROR";

            flag =1;

            return 0;

        }

        h[i]=h[i]-d[i];

        max=-1;

        for(int j=0; j<4; j++){

            if(h[j]>max){

            max=h[j];

        }

        }

        if(max>=2\*h[i]){

            rez = "NO";

        }

    }

    if (flag == 0){

        cout<<rez;

    }

}

Першим етапом роботи є під’єднання потрібних бібліотек, у нашому випадку це «iostream» та «string». Далі оголошуємо змінні у потрібних нам типах даних та вводимо значення масиву h[i]. Під час введення визначаємо максимальний елемент даного масиву. Після, вводимо масив d[i] та переходимо до основної частини програми. В циклі for, маємо декілька операторів розгалуження. Перший if перевіряє чи не більша довжина, яку ми хочемо відрізати, від довжини ніжки столу і якщо це так, то ми виходимо з циклу та надаємо прапорцю значення 1. Після, ми «відрізаємо» частини ніжок столу і змінюємо значення максимальної висоти ніжки столу в циклі for, в наступному операторі if перевіряємо чи буде стіл зберігати положення паралельно підлозі. Після виходу з циклу перевіримо чи наш прапорець досі дорівнює нулю і, якщо це так, то виведемо «YES».

**Завдання №4**

#include<iostream>

#include<string>

using namespace std;

int main(){

    int n, vasia=0, kolia=0, k=0, v=0;

    cin>>n;

    string c;

    cin>>c;

    for(int i=0; i<n; i++){

        if( c[i]=='V'){

            vasia++;

        }

        else {

            kolia++;

        }

        if((kolia>10)&&(kolia>vasia+1)){

            k++;

            kolia=0;

            vasia=0;

        }

        else if((vasia>10)&&(vasia>kolia+1)){

            v++;

            kolia=0;

            vasia=0;

        }

    }

    cout<<k<<":"<<v<<endl;

    if(kolia!=0 or vasia!=0){

    cout<<kolia<<":"<<vasia;

    }

    return 0;

}

Перш за все, під’єднаємо бібліотеки «iostream» та «string». Після, ініціалізуємо потрібні змінні та масив та вводимо їх значення. Далі в циклі for перебираємо значення введеного масиву та в залежності від букви підраховуємо рахунок в одній партії (перші два оператори розгалуження), після рахуємо хто з хлопців скільки партій переміг(наступні 2 оператори розгалуження). Вийшовши з циклу виводимо рахунок по партіях і якщо хлопці не дограли останню гру – виводимо і її рахунок.

**Завдання №5**

#include<iostream>

#include<string>

using namespace std;

int main(){

    string sunny="sunny";

    string rainy="rainy";

    string cloudy="cloudy";

    string snowy="snowy";

    string windy="windy";

    string weather;

    cout<<"Enter weather tipe:\n";

    cin >> weather;

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

        cout<<"It's better to wear a jacket\n\n";

    } else{

        cout<<"It`s warm outside, you don`t need a jacket\n\n";

    }

    if(weather==sunny || weather == cloudy){

        cout<<"You can go for a walk\n\n";

    } else{

        cout<<"You can visit your friends\n\n";

    }

    switch (weather[0])

        {

        case 'r':

        cout <<"It is better to choose waterproof shoes";

        break;

        case 'w' || 'c':

        cout<<"It is better to choose sneakers or some warmer";

        break;

        case 's':

        if(weather[1]=='n'){

            cout<<"It is better to choose warm shoes";

        } else{

            cout<<"It is better to choose something like convers or loafers";

        }

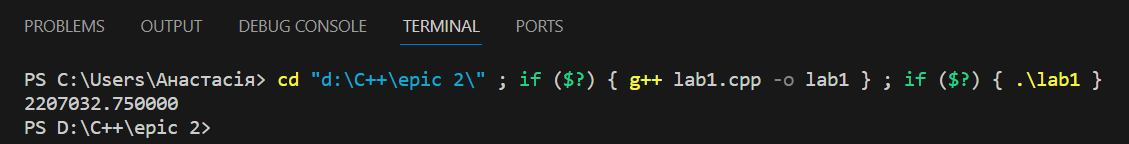
    }

}

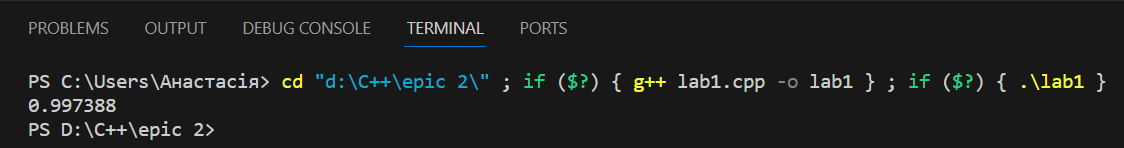
Підключаємо потрібні бібліотеки, а саме «iostream» та «string». Наступним етапом є оголошення усіх змінних та введення значення weather. Після, за допомогою оператора if, в залежності від значення, яке було введено, виводимо необхідні рекомендації що до одягнення куртки та активності. Потім, використовуючи оператор switch даємо рекомендації що до взуття. Оскільки в нас 2 слова починаються на одну і ту ж букву, то потрібно додатково використати if і перевірити 2-гу літеру.

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

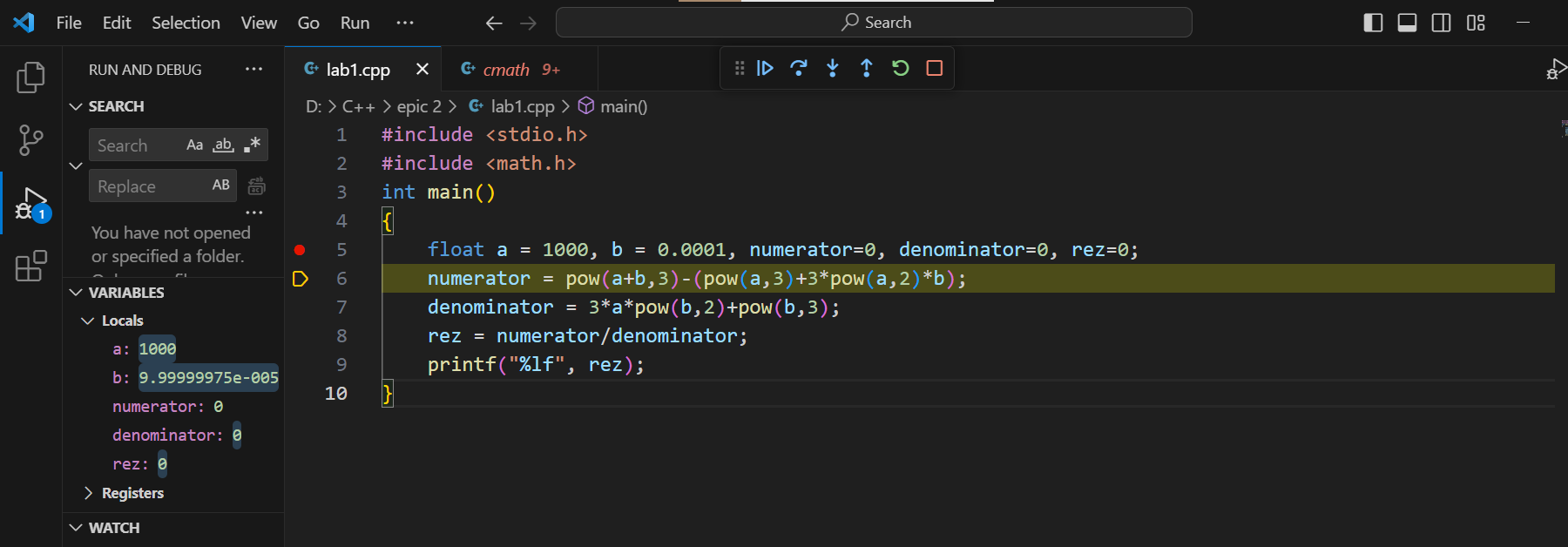
**Завдання №1**



Тип даних float



Тип даних double

****

Чому з типом даних float не правильна відповідь

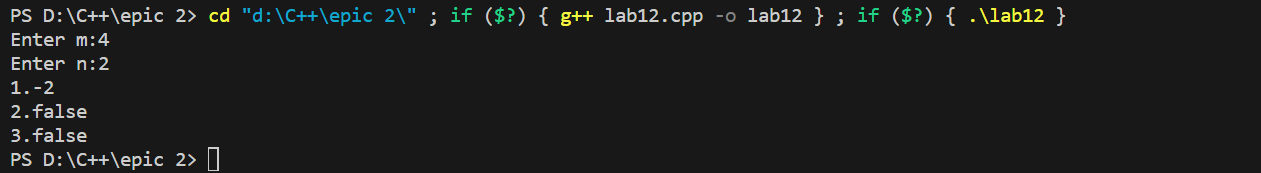
Під час обчислення з типом даних float програма видає неправильне значення, адже число b з самого початку не поміщається в цей тип даних і замість нього ми можемо бачити надзвичайно велике значення.

Затрачений час: 4 години

**Завдання №2**

****

Тест №1



Тест №2

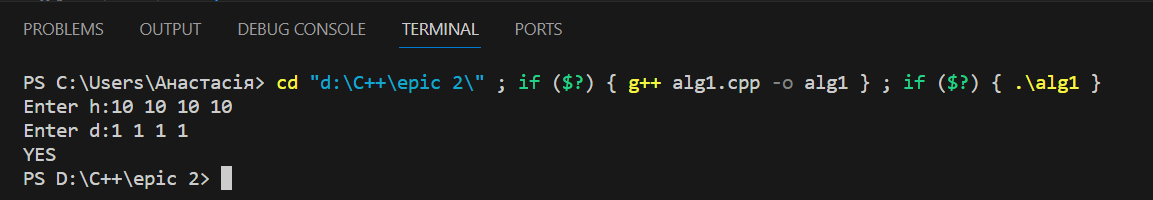


Тест №3

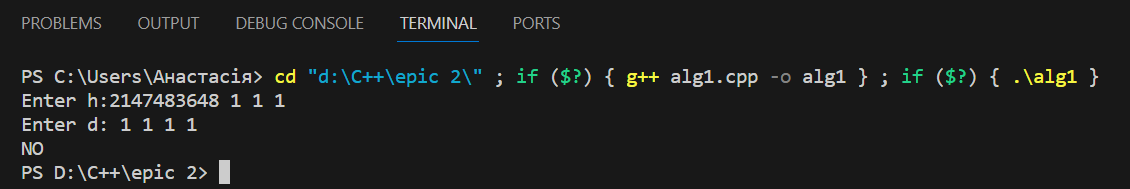
Такі результати пов’язані з тим, що знаки інкременту і декременту по пріоритетності зчитування йдуть після знаків порівняння та віднімання.

Затрачений час: 40 хв

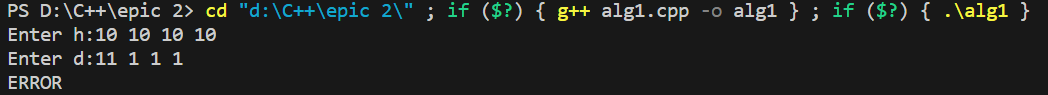
**Завдання №3**

****

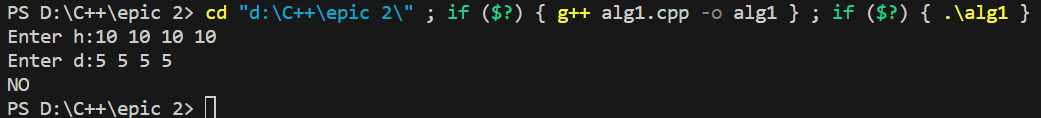
Тест №1



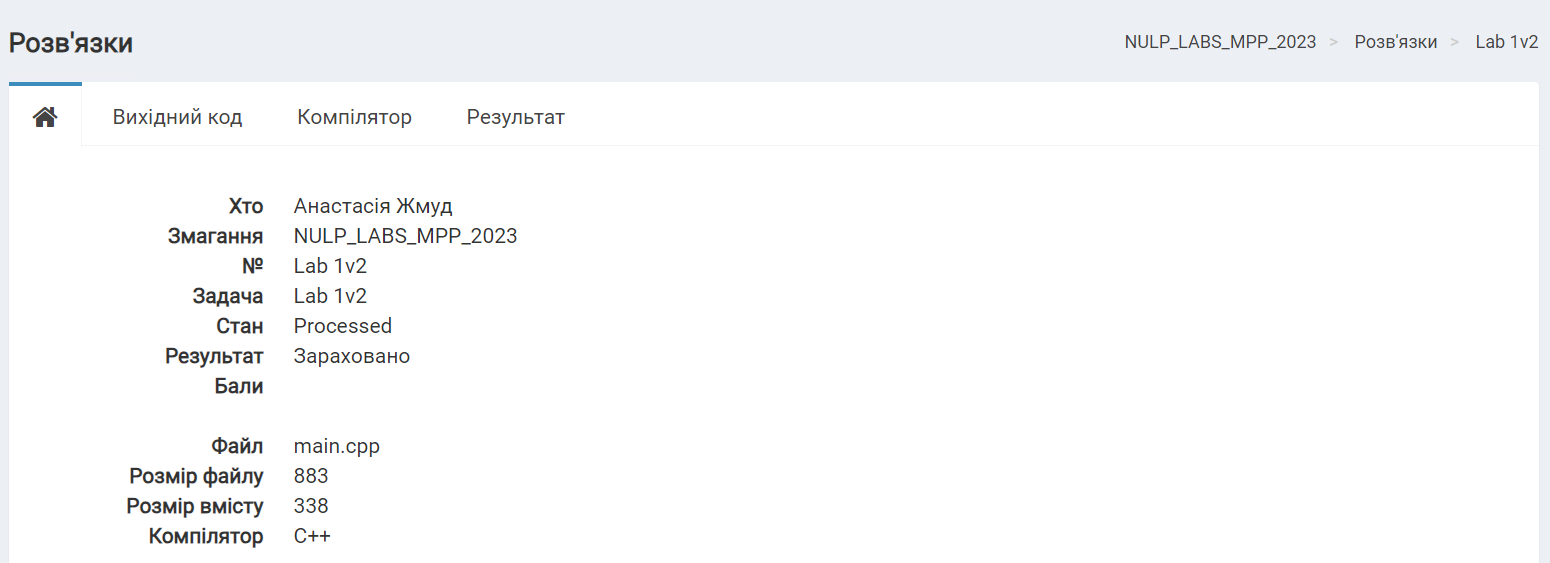
Тест №2



Тест №3



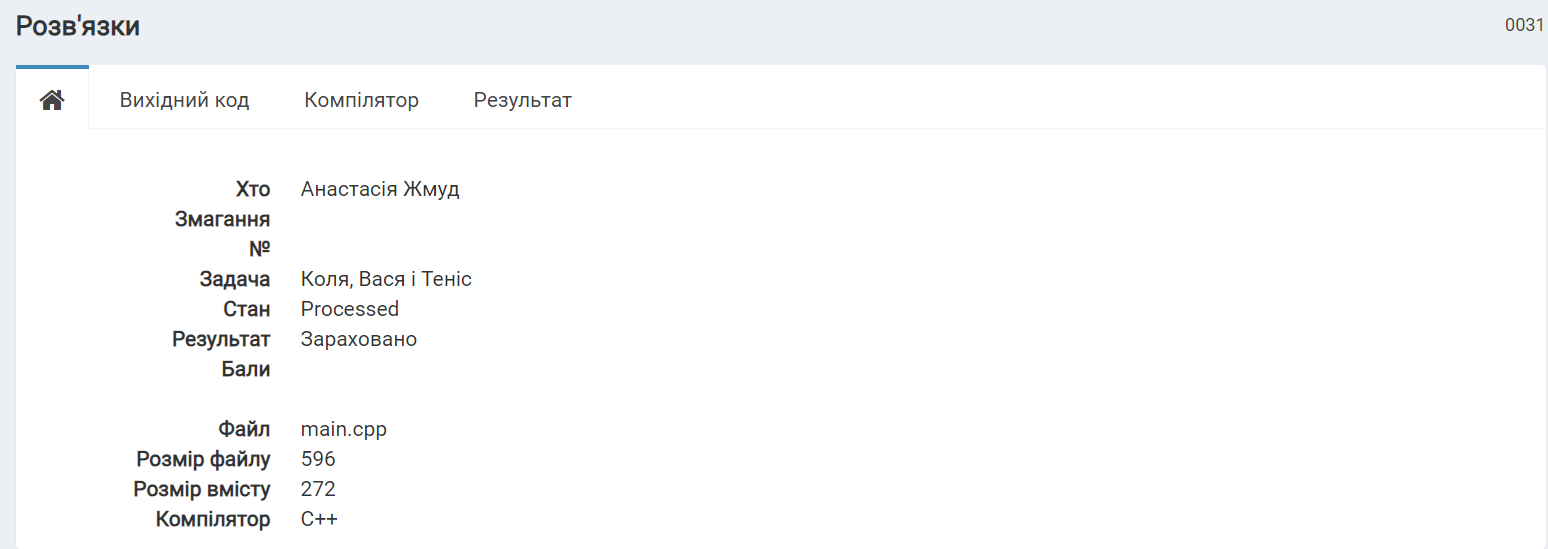
Тест №4



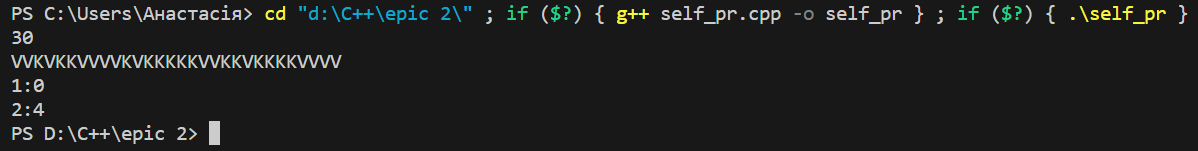
Algotester зарахування

Затрачений час: 4 години

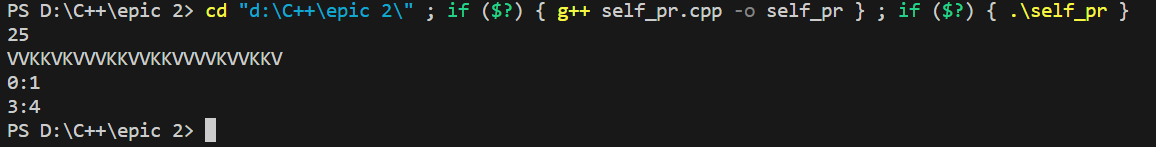
**Завдання №4**

****

Algotester зарахування

****

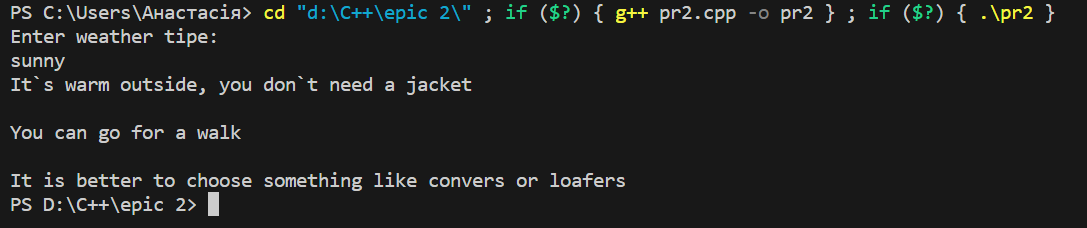
Тест №1



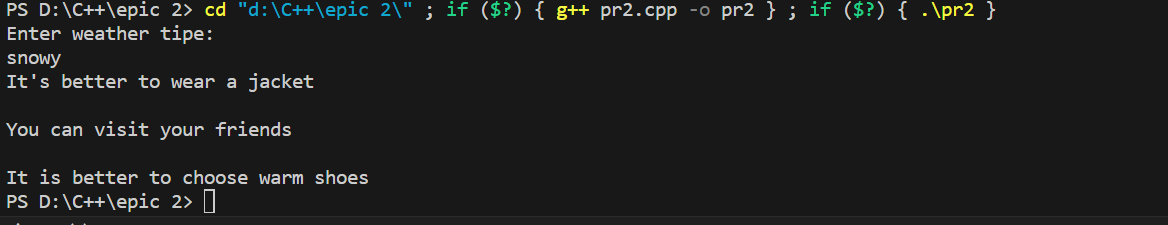
Тест №2

Затрачений час: 3 години

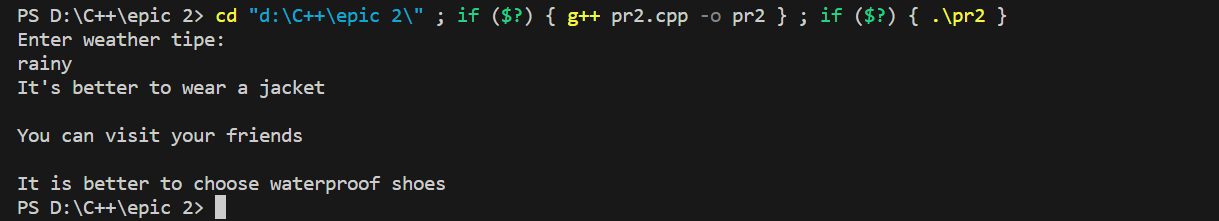
**Завдання № 5**

****

Тест №1



Тест №2



Тест №3

Затрачений час: 1 година

# **Висновки:**

Працюючи над розділом №2 та виконуючи практичні та лабораторні роботи я ознайомилась з різними типами даних та їх областю значень. Зрозуміла важливість врахування типу даних та пріоритетності зчитування знаків під час написання коду. Ознайомилася з алгоритмами та операторами циклу й розгалуження. Набула навичок роботи з іншими різноманітними операторами C та C++ та бібліотеками для цих мов.