Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра систем штучного інтелекту



**Звіт**

про виконання

**Лабораторних та практичних робіт № 1, 2**

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

***з розділу***: «Лінійні та розгалужені алгоритми. Умовні оператори. Константи, змінні»

***Виконав:***

студент групи ШІ-12

Яцишин Ігор Васильович

# **Тема роботи:**

Константи та змінні в мові C++, робота з ними. Лінійні алгоритми та розгалужені алгоритми if/else if/else та switch. Оператор повторення for.

# **Мета роботи:**

Навчитися працювати з константами та змінними в мові C++. Опрацювати та закріпити практично роботу з операторами галуження та повторення.

# **Теоретичні відомості:**

1. Теоретичні відомості з переліком важливих тем:

* Тема №1: Константи та змінні в мові C++.
* Тема №2: Умовні оператори в C++.
* Тема №3: Оператор switch в мові C++.
* Тема №4: Цикл for в мові C++.

1. Індивідуальний план опрацювання теорії:

* Тема №1: Константи та змінні в мові C++.
  + Джерела Інформації
    - Стаття: C++ Variables, Literals and Constants

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

* + Що опрацьовано:
    - Опрацьовано декларування констант, оголошення змінних та принцип роботи з ними
  + Статус: Ознайомлений
  + Початок опрацювання теми: 14.10.2023 ~20.00
  + Звершення опрацювання теми: 14.10.2023 ~21.30
* Тема №2: Умовні оператори в C++.
  + Джерела Інформації:
    - Стаття: C++ Conditions and If Statements <https://www.w3schools.com/cpp/cpp_conditions.asp>
    - Відео: [Урок #5] C++ Для новачків. Розгалуження <https://www.youtube.com/watch?v=3W4DEm8EG6Q>
  + Що опрацьовано:
    - Опрацьовано теоретичний матеріал про умовні оператори розгалуження, а саме if, else if, else та “short hand” if..else для подальшого застосування на практиці
  + Статус: Ознайомлений
  + Початок опрацювання теми: 16.10.2023 ~16.30
  + Звершення опрацювання теми: 16.10.2023 ~ 20.15
* Тема №3: Оператор switch в мові C++.
  + Джерела Інформації:
    - Відео: C++ Теорія ⦁Урок 30 ⦁switch <https://www.youtube.com/watch?v=OuPiJ5SMYis>
    - Стаття: C++ Switch <https://www.w3schools.com/cpp/cpp_switch.asp>
  + Що опрацьовано:
    - Опрацьовано теоретичний матеріал щодо роботи оператора switch, конструкцій break та default. Розглянути аспекти роботи з даними конструкціями
  + Статус: Ознайомлений
  + Початок опрацювання теми: 17.10.2023 ~23.00
  + Звершення опрацювання теми: 18.10.2023 ~00.10.
* Тема №4: Цикл for в мові C++.
  + Джерела Інформації:
    - Стаття: C++ for Loop <https://www.programiz.com/cpp-programming/for-loop>
  + Що опрацьовано:
    - Вивчено теоретичний матеріал щодо роботи циклу for в мові C++, його синтаксису та прикладів застосування. Також ознайомився з побудовою циклу for в FlowCharts
  + Статус: Ознайомлений
  + Початок опрацювання теми: 22.10.2023 ~14.50
  + Звершення опрацювання теми: 22.10.2023 ~18.00

# **Виконання роботи:**

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

Завдання №1 VNS Lab 1 Task 1

* Варіант завдання - 1
* Деталі завдання : обчислити значення виразу на основі заданих значень констант
* Важливі деталі для врахування в імплементації програми:

Використання функції exp() для обчислення значення експоненти в певному степені та функції pow() для піднесення чисел до степеня

Завдання №2 VNS Lab 1 Task 2

* Варіант завдання - 1
* Деталі завдання : пояснити значення виразів згідно з варіантом
* Важливі деталі для врахування в імплементації програми:

Ввивід резултату виконання програми

Завдання №3 VNS Lab 2 Task 1

* Варіант завдання - 25
* Деталі завдання : обчислити суму перших семи членів ряду
* Важливі деталі для врахування в імплементації програми:

Використання циклу for для обчислення суми кількох членів ряду

Завдання №4 Algotester Lab 1

* Варіант завдання - Task 2
* Деталі завдання : <https://drive.google.com/file/d/1lRdmsfnUHCf2SynACnqAXseGmrBWwYvS/view?usp=sharing>
* Важливі деталі для врахування в імплементації програми:

Використання умовних операторів розгалуження if/else if/ else

Завдання №5 Algotester Lab 2

* Варіант завдання - Task 1
* Деталі завдання : <https://drive.google.com/file/d/1JBYiTaA6IdCdZ0XlU5VQQaifaHlkBriQ/view?usp=sharing>
* Важливі деталі для врахування в імплементації програми:

Використання умовних операторів розгалуження if/else if/ else, робота з динамічним масивом

Завдання №6 Class Practice Work

* Варіант завдання - немає
* 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:

Використання наступної структури:

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 №7 Self Practice Work

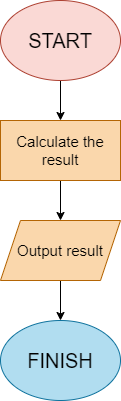
* Варіант завдання - немає
* Деталі завдання : <https://algotester.com/uk/ArchiveProblem/DisplayWithEditor/6>
* Важливі деталі для врахування в імплементації програми:

Введення цілочисельного значення змінної n для правильності роботи; використання умовних операторів галуження

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

Програма №1 VNS Lab 1 Task 1

* Блок-схема

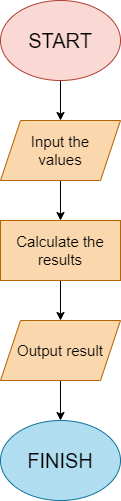


*Рисунок 1. Блок-схема програми №1*

* Планований час на реалізацію: 35 хвилин

Програма №2 VNS Lab 1 Task 2

* Блок-схема

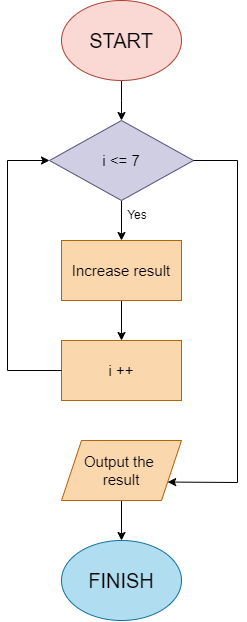


*Рисунок 2. Блок-схема програми №2*

* Планований час на реалізацію: 20 хвилин

Програма №3 VNS Lab 2 Task 1

* Блок-схема

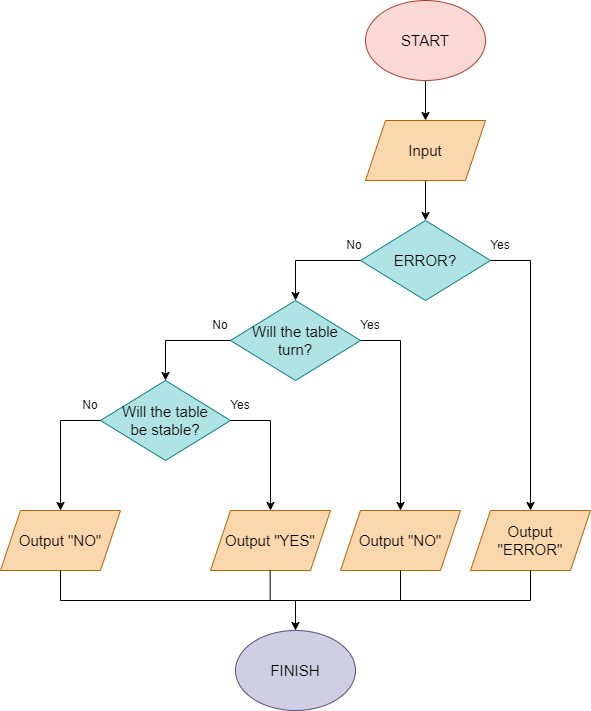


*Рисунок 3. Блок-схема програми №3*

* Планований час на реалізацію: 35 хвилин

Програма №4 Algotester Lab 1 Task 2

* Блок-схема

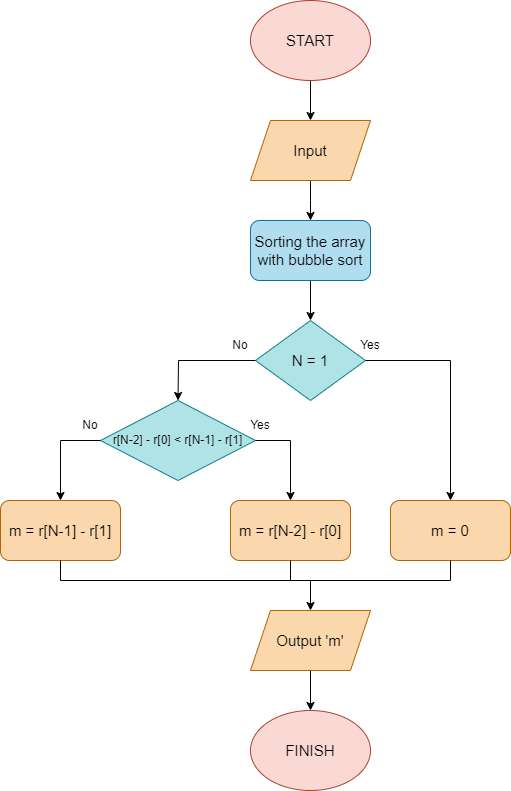


*Рисунок 4. Блок-схема програми №4*

* Планований час на реалізацію: 60 хвилин

Програма №5 Algotester Lab 2 Task 1

* Блок-схема

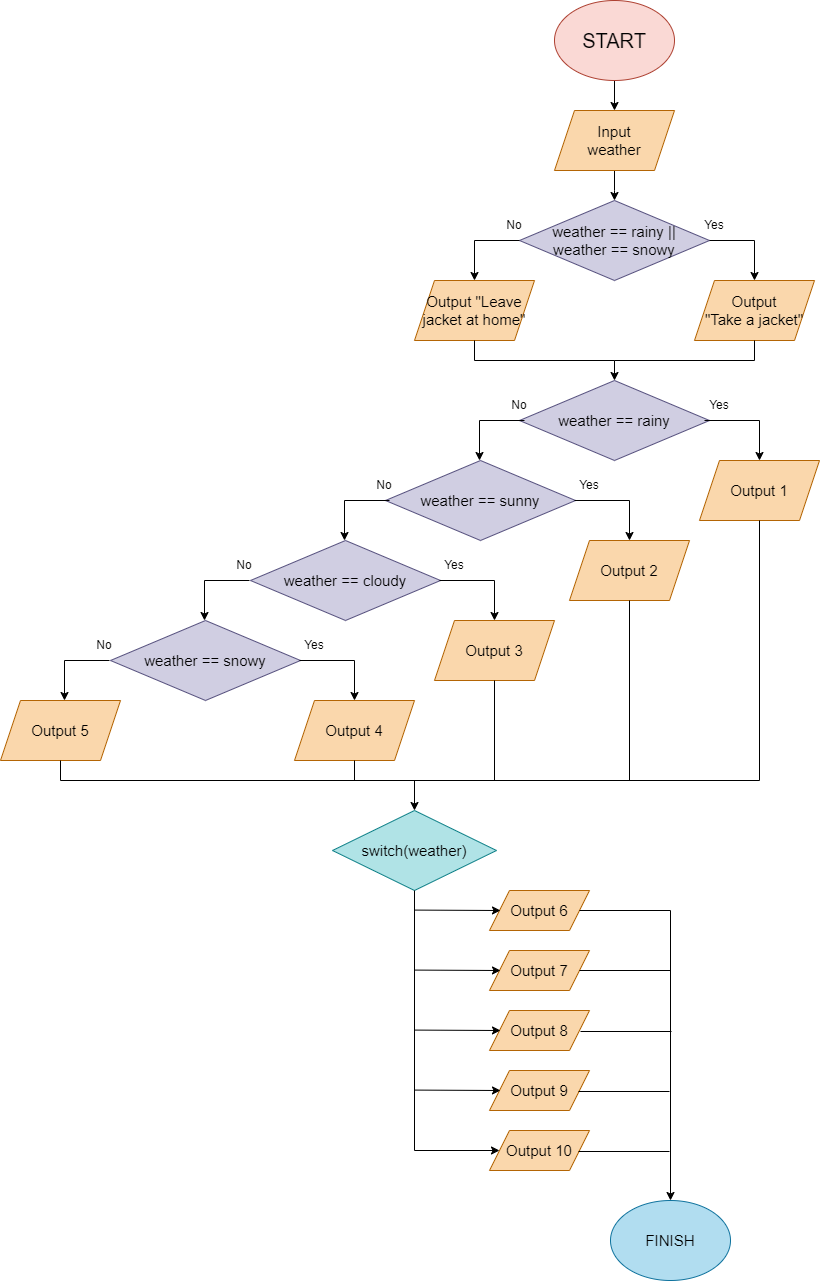


*Рисунок 5. Блок-схема програми №5*

* Планований час на реалізацію: 80 хвилин

Програма №6 Class Practice Work

* Блок-схема

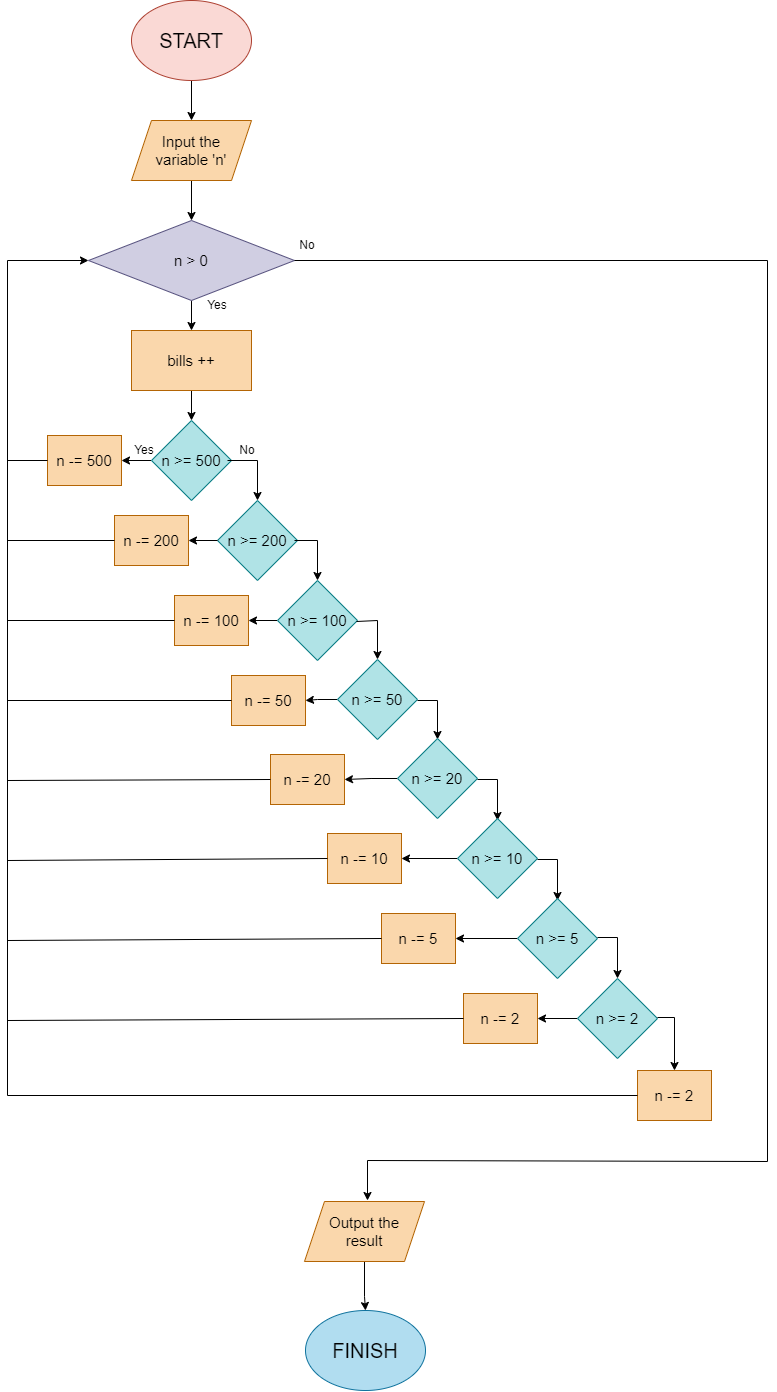


*Рисунок 6. Блок-схема програми №6*

* Планований час на реалізацію: 80 хвилин

Програма №7 Self Practice Work

* Блок-схема



*Рисунок 7. Блок-схема програми №7*

* Планований час на реалізацію: 50 хвилин

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

Додаткова конфігурація середовища для роботи над поставленими завданнями не проводилася.

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

Завдання №1 VNS Lab 1 Task 1 Variant 1

#include <cmath>

#include <cstdio>

#define a 1000

#define b 0.0001

*int* main() {

*float* res;

    res = (pow(a + b, 2) - (pow(a, 2) + 2\*a\*b))/(pow(b, 2));

    printf("%f", res);

    return 0;

}

Програма виконує математичні обрахунки результату, що заданий відповідною формулою. Результат виводиться на екран функцією printf.

Посилання на файл програми у пул-запиті GitHub

<https://github.com/artificial-intelligence-department/ai_programming_playground/pull/112>

Завдання №2 VNS Lab 1 Task 2 Variant 1

#include <cstdio>

*int* main() {

*int* n, m;

    printf("Enter n: ");

    scanf("%d", &n);

    printf("Enter m: ");

    scanf("%d", &m);

    printf("%d \n", n+++m);

    printf("%d \n", m-->n);

    printf("%d \n", n-->m);

    printf("%d", n);

    return 0;

}

Програма отримує цілочисельні значення змінних n та m та проводить низку операцій. Спершу до числа n додають m та постфіксно інкрементують n. Наступним кроком роблять перевірку m > n та постфіксно декрементують m. Після цього проводиться перевірка n > m та постфіксно декрементується n. Усі результати виводяться на екран функцією printf (в 2 та 3 порівняннях у виводі 0 = false; 1 = true)

Посилання на файл програми у пул-запиті GitHub

<https://github.com/artificial-intelligence-department/ai_programming_playground/pull/112>

Завдання №3 VNS Lab 2 Task 1 Variant 25

##include <cstdio>

#include <math.h>

*int* main() {

*double* res = 0;

    for(*int* n = 0; n <= 7; ++n){

        res += pow(n, 2) \* exp(-pow(n, 0.5));

    }

    printf("The summ of the series equals %lf.", res);

    return 0;

}

Цикл фор на кожну ітерацію обчислює елемент an ,починаючи з першого і до сьомого, і додає його до результату. Кінцевий результут функцією printf виводиться в консоль

Посилання на файл програми у пул-запиті GitHub

<https://github.com/artificial-intelligence-department/ai_programming_playground/pull/112>

Завдання №4 Algotester Lab 1 Task 2

#include <iostream>

#include <algorithm>

*int* main() {

*long* *long* h1, h2, h3, h4, d1, d2, d3, d4;

    std::cin >> h1 >> h2 >> h3 >> h4;

    std::cin >> d1;

    std::cin >> d2;

    std::cin >> d3;

    std::cin >> d4;

    if (h1 < d1 || h2 < d2 || h3 < d3 || h4 < d4) {

        std::cout << "ERROR"<< std::endl;

    }else if (2 \* std::min({h1-d1, h2, h3, h4}) <= std::max({h2, h3, h4})) {

        std::cout << "NO" << std::endl;

    }else if (2 \* std::min({h2-d2, h3, h4}) <= std::max({h1-d1, h3, h4})) {

        std::cout << "NO" << std::endl;

    }else if(2 \* std::min(h3-d3, h4) <= std::max({h1-d1, h2-d2, h4})) {

        std::cout << "NO" << std::endl;

    }else if(2 \* (h4-d4) <= std::max({h1-d1, h2-d2, h3-d3})) {

        std::cout << "NO" << std::endl;

    }else if (h1-d1 == h2-d2 && h2-d2 == h3-d3 && h3-d3 == h4-d4 && std::min({h1-d1, h2-d2, h3-d3, h4-d4}) != 0){

        std::cout << "YES" << std::endl;

    }else {

        std::cout << "NO" << std::endl;

    }

    return 0;

}

Програма отримує цілочисельні значення змінних h1, h2, h3, h4, d1, d2, d3, d4. Розгалуженнями if-else if-else перевіряються на виконяння умови положення стола (“YES” - стіл паралельний підлозі; “NO” - стіл перевернувся; “ERROR” - помилка)

Посилання на файл програми у пул-запиті GitHub

<https://github.com/artificial-intelligence-department/ai_programming_playground/pull/112>

Завдання №5 Algotester Lab 2 Task 1

#include <iostream>

#include <algorithm>

*void* bubbleSort(*long* *arr*[], *long* *size*);

*int* main() {

*long* N, m;

    std::cin >> N;

*long* r[N];

    for(*int* i = 0; i < N; i++) {

        std::cin >> r[i];

    }

    bubbleSort(r, N);

    if (N == 1){

        m = 0;

    }else if(r[N - 2] - r[0] < r[N - 1] - r[1]) {

        m = r[N - 2] - r[0];

    }else {

        m = r[N - 1] - r[1];

    }

    std::cout << m << std::endl;

    return 0;

}

*void* bubbleSort(*long* *arr*[], *long* *size*) {

    for(*int* i = 0; i < *size*; ++i){

        for(*int* j = 0; j < *size* - i; ++j){

            if (*arr*[j] > *arr*[j+1]){

*int* temp = *arr*[j];

*arr*[j] = *arr*[j+1];

*arr*[j+1] = temp;

            }

        }

    }

}

Програма отримує цілочисельне значення змінної N та кожнен елемент масиву r[] розміром N елементів. Шляхом “видалення” першого або останнього значеня масиву r розгалуження if мінімізує різницю найбільшого і найменшого елемента. Результат виводиться в консоль

Посилання на файл програми у пул-запиті GitHub

<https://github.com/artificial-intelligence-department/ai_programming_playground/pull/112>

Завдання №6 Class Practice Work

#include <iostream>

#include <string>

*int* main() {

    std::string weather;

input:

    std::cout << "Choose the weather from suggested (sunny; rainy; cloudy; snowy; windy): ";

    std::cin >> weather;

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

        goto input;

    }

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

        std::cout << "You should better take a jacket \n";

    }else{

        std::cout << "It's warm enough, leave your jacker at home! \n";

    }

    if(weather == "rainy"){

        std::cout << "You can spend your evening watching a film) \n";

    }else if(weather == "sunny"){

        std::cout << "Take your bike and go for a ride \n";

    }else if(weather == "cloudy"){

        std::cout << "Don't forget to meet your friends \n";

    }else if(weather == "snowy"){

        std::cout << "Nice time to play snow balls! \n";

    }else if(weather == "windy"){

        std::cout << "You'd rather stay home and thy to cook some new dishes \n";

    }

    switch (weather[0]) {

        case 's':

            if(weather == "sunny"){

                std::cout << "Wear your snikers or sandales \n";

            }else{

                std::cout << "Wear your warm boots \n";

            }

            break;

        case 'r':

            std::cout << "Wear your rainy boots \n";

            break;

        case 'c':

            std::cout << "Wear anything you want)) \n";

            break;

        case 'w':

            std::cout << "Wear some sturdy shoes \n";

            break;

    }

    return 0;

}

Програма отримує від користувача значення погоди типу string. При неправильному введенні користувач змушений виконати ввечення заново. На основі вводу програма надає рекомендації щодо верхнього одягу, активностей та взуття.

Посилання на файл програми у пул-запиті GitHub

<https://github.com/artificial-intelligence-department/ai_programming_playground/pull/112>

Завдання №7Self Practice Work

#include <iostream>

*int* main() {

*int* n, bills = 0;

    std::cin >> n;

    while(n > 0){

        bills += 1;

        if(n >= 500){

            n -= 500;

        }else if (n >= 200){

            n -= 200;

        }else if(n >= 100){

            n -= 100;

        }else if(n >= 50){

            n -= 50;

        }else if (n >= 20){

            n -= 20;

        }else if (n >= 10){

            n -= 10;

        }else if(n >= 5){

            n -= 5;

        }else if(n >= 2){

            n -= 2;

        }else {

            n -= 1;

        }

    }

    std::cout << bills << std::endl;

    return 0;

}

Програма отримує від користувача значення певної суми n типу integer. На основі вводу програма шукає скільки мінімум купюр знадобиться, щоб отримати певну суму. Результат виконання виводиться в консоль.

Посилання на файл програми у пул-запиті GitHub

<https://github.com/artificial-intelligence-department/ai_programming_playground/pull/112>

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

Завдання №1 VNS Lab 1 Task 1 Variant 1

OUTPUT:

1.001172

Час затрачений на виконання завдання: 30 хвилин

Завдання №2 VNS Lab 1 Task 2 Variant 1

INPUT:

Enter n: 3

Enter m: 4

OUTPUT:

7

0

1

3

Час затрачений на виконання завдання: 25 хвилин

Завдання №3 VNS Lab 2 Variant 25

OUTPUT:

The summ of the series equals 14.354754.

Час затрачений на виконання завдання: 30 хвилин

Завдання №4 Algotester Lab 1 Task 2

INPUT:

10 10 10 10

5

5

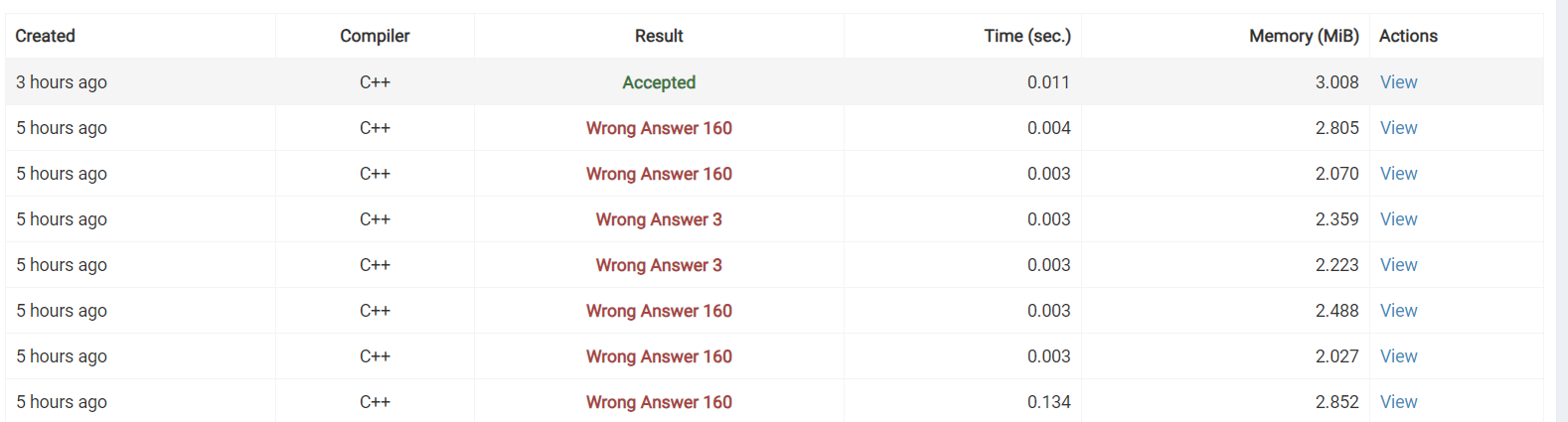
5

5

OUTPUT:

NO

Час затрачений на виконання завдання: 95 хвилин



*Рисунок 8. Результат успішного виконання програми №4*

Завдання №5 Algotester Lab 2 Task 1

INPUT:

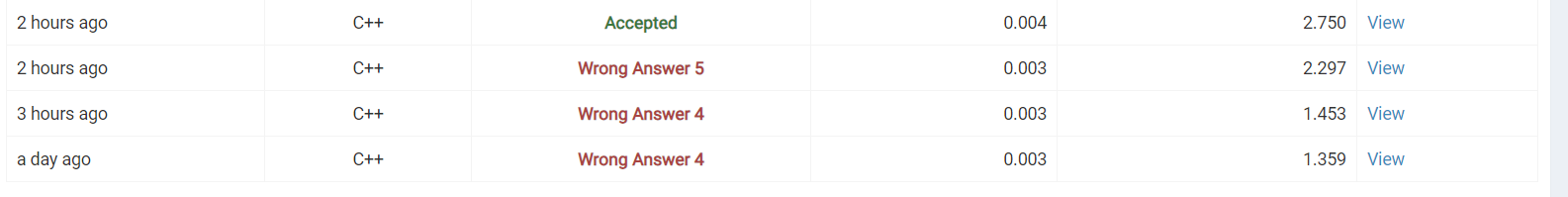
5

1 2 2 4 4

OUTPUT:

2

Час затрачений на виконання завдання: 80 хвилин



*Рисунок 9. Результат успішного виконання програми №5*

Завдання №6 Class Practice Work

INPUT:

Choose the weather from suggested (sunny; rainy; cloudy; snowy; windy): rainy

OUTPUT:

You should better take a jacket

You can spend your evening watching a film)

Wear your rainy boots

Час затрачений на виконання завдання: 70 хвилин

Завдання №7 Self Practice Work

INPUT:

74

OUTPUT:

4

Час затрачений на виконання завдання: 45 хвилин

# **Висновки:**

Під час виконання лабораторних робіт №1, 2 (ВНС) я навчився працювати зі змінними в мові C++, також запрограмував цикл повторень. Практичні завдання та лабораторні на платформі Алготестер допомогли закріпити набуті навички та знання на задачах більш прикладного характеру. Вважаю, що зможу оперувати галуженнями та циклами в подальшому застосовуючи їх на практиці.