

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. Змінні. Константи. Типи даних. Розмір типів даних. Ввід вивід. Базові операції та вбудовані функції. Коментарі.»

з дисципліни: «Основи програмування»

до:

ВНС Лабораторної Роботи № 1
Алготестер Лабораторної Роботи № 1
Практичних Робіт до блоку № 2

Виконав:

Студент групи ШІ-12
Кривичко Назар Русланович

Львів 2024

Тема роботи:

Лабораторні завдання:

- а. Виконання програмування в рамках VNS Lab 1, завдання 1 і 2.
- б. Завдання програмування в Algotester Lab 1, завдання 1.

Мета роботи:

Ознайомлення з основами програмування через вивчення лінійних та розгалужених алгоритмів, а також застосування умовних та логічних операторів. Використання змінних та констант, розуміння типів даних і їх розмірів, а також вивчення принципів вводу/виводу, базових операцій і вбудованих функцій.

Теоретичні відомості:

У даній роботі розглядаються основні концепції мови програмування C++, зокрема, потоки вводу та виводу, унарні та бінарні оператори, бітові оператори, а також умовні конструкції, такі як `if-else` та `switch-case`, а також основні лінійні алгоритми.

Джерела :

книга - Stephen Prata - “ *C++ Primer Plus* ”

книга - Aditya Y.Bhargava - “ *Grokking algorithms* ”

Виконання роботи:

Завдання № 1

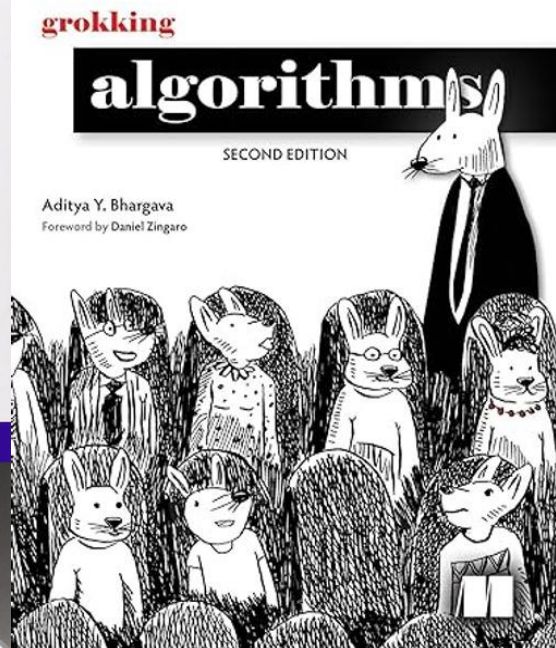
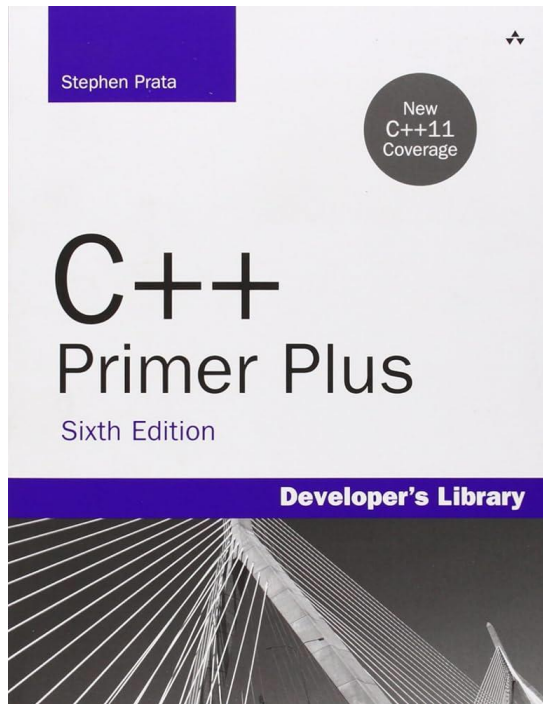
Requirements :

Theory Education Activities

Time:

Expected: 2-3 weeks

Spent: up to 2 month



Завдання № 2

Requirements :

Requirements management (understand tasks) and design activities (draw flow diagrams and estimate tasks 3-7)

Time:

Expected: 1 hour

Spent: ~ 1 hour

Завдання № 3

Requirements :

VNS Lab 1 Task 1

Time:

Expected: 1 hour | Spent: ~ 20 mins

```

#include<iostream>
#include<math.h>
int main(void)
{
    //For float
    // (a + b)^3 - a^3 / 3ab^2 + b^3 + 3a^2b
    float a = 1000, b = 0.0001;
    float upperPart = (pow((a + b), 3) - pow(a, 3));
    float lowerPart = (3 * a * pow(b, 2) + pow(b, 3) + 3 * pow(a, 2) * b);
    float result = upperPart / lowerPart;

    //For double

    double aD = 1000, bD = 0.0001;
    double upperPartD = (pow((aD + bD), 3) - pow(aD, 3));
    double lowerPartD = (3 * aD * pow(bD, 2) + pow(bD, 3) + 3 * pow(aD, 2) * bD);
    double resultD = upperPartD / lowerPartD;

    std::cout << "For float: " << result << std::endl
                << "For double: " << resultD << std::endl;

    return 0;
}

```

Explanation:

Різні результати в обчисленнях для float і double виникають через різницю в точності представлення чисел у пам'яті: float має меншу точність (приблизно 7 значущих цифр), ніж double (близько 15-16 значущих цифр)

Завдання № 4

Requirements :

VNS Lab 1 Task 2

Time:

Expected: 10 mins

Spent: ~ 10 mins

```
#include<iostream>

int main(void)
{
    int n, m;
    std::cin >> n >> m;

    std::cout << n++ * m << std::endl;
    std::cout << std::boolalpha;
    std::cout << (n++ < m) << std::endl;
    std::cout << (m-- > m) << std::endl;

    return 0;
}
```

Explanation:

Різні результати в обчисленнях з n та m в цьому коді пов'язані з порядком виконання операторів інкременту ($++$) і декременту ($--$), а також з їх впливом на значення змінних. Наприклад, в першому рядку `std::cout << n++ * m` використовується значення n до його інкременту, тоді як в другому рядку `std::cout << (n++ < m)` - значення n вже збільшене (лише 1 раз з минулого), що може призвести до різних результатів при порівнянні з m .

Завдання № 5

Requirements :

Algotester Lab 1 Task 1

Time:

Expected: 1 hour

Spent: ~ 20 mins

```
#include <iostream>
using namespace std;

int main() {
    long long cubes[5];

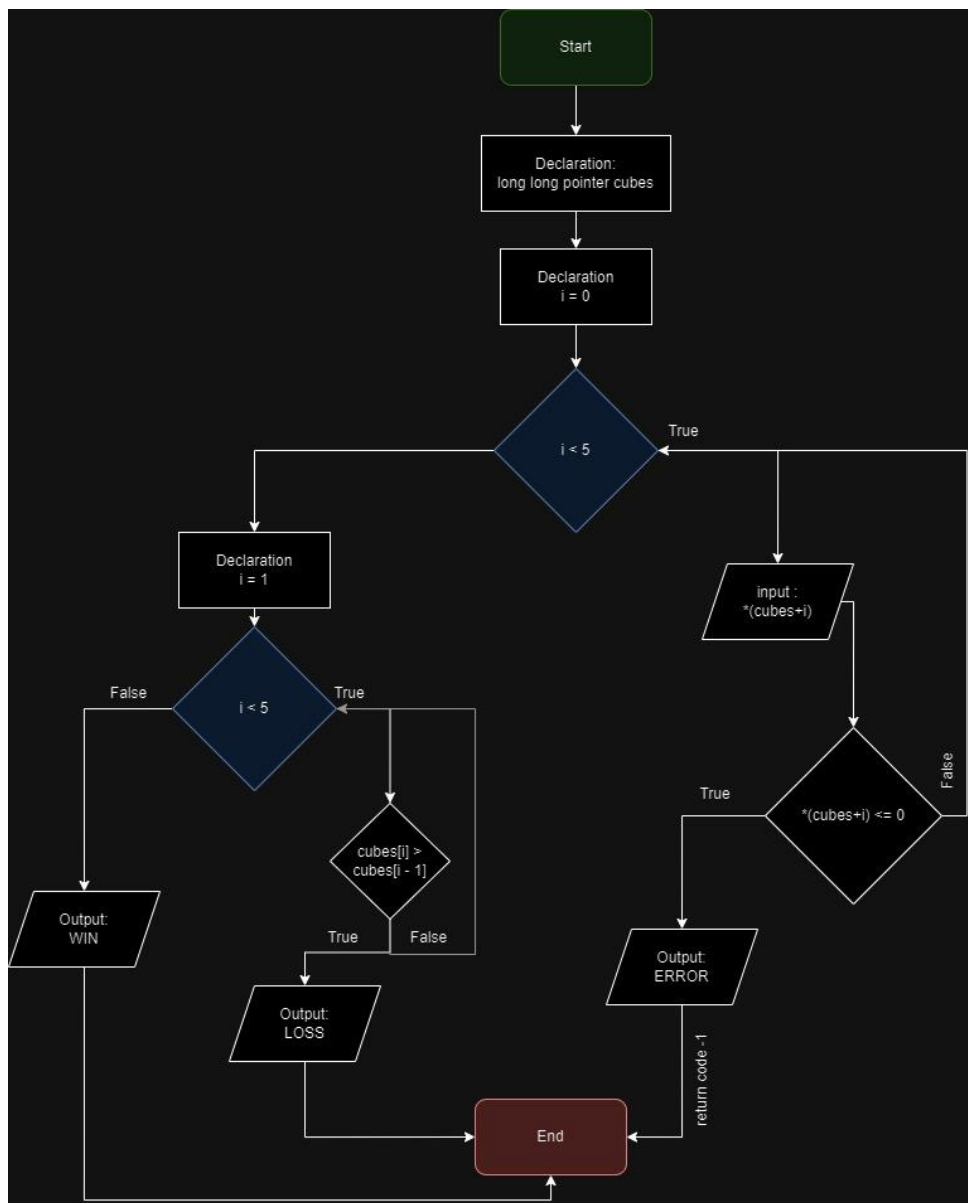
    for (int i = 0; i < 5; i++) {
        cin >> cubes[i];

        if (cubes[i] ≤ 0 ) {
            cout << "ERROR" << endl;
            return 0;
        }
    }

    for (int i = 1; i < 5; i++) {
        if (cubes[i] > cubes[i - 1]) {
            cout << "LOSS" << endl;
            return 0;
        }
    }

    cout << "WIN" << endl;
    return 0;
}
```

Flowchart Diagram:



Завдання № 6

Requirements :

Class Practice Task

Time:

Expected: 20 mins

Spent: ~ 20 mins

```

#include <iostream>
#include <string>
#define ERR_C 1
using namespace std;

int main() {
    string weather;

    cout << "Введіть поточні погодні умови (sunny, rainy, cloudy, snowy, windy): ";
    cin >> weather;

    if (weather != "sunny" && weather != "rainy" && weather != "cloudy" && weather != "snowy" && weather != "windy")
        cout << "Будь ласка, введіть дійсну умову погоди!" << endl;
        exit(ERR_C);
    }

    if (weather == "snowy" || weather == "rainy") {
        cout << "Не забудьте взяти куртку!" << endl;
    }
    else {
        cout << "Куртка не потрібна." << endl;
    }
    if (weather == "sunny") {
        cout << "Чудовий день для пікніка!" << endl;
    }
    else if (weather == "rainy") {
        cout << "Ідеальна погода, щоб читати книгу всередині!" << endl;
    }
    else if (weather == "cloudy") {
        cout << "Може, відвідати музей?" << endl;
    }
    }

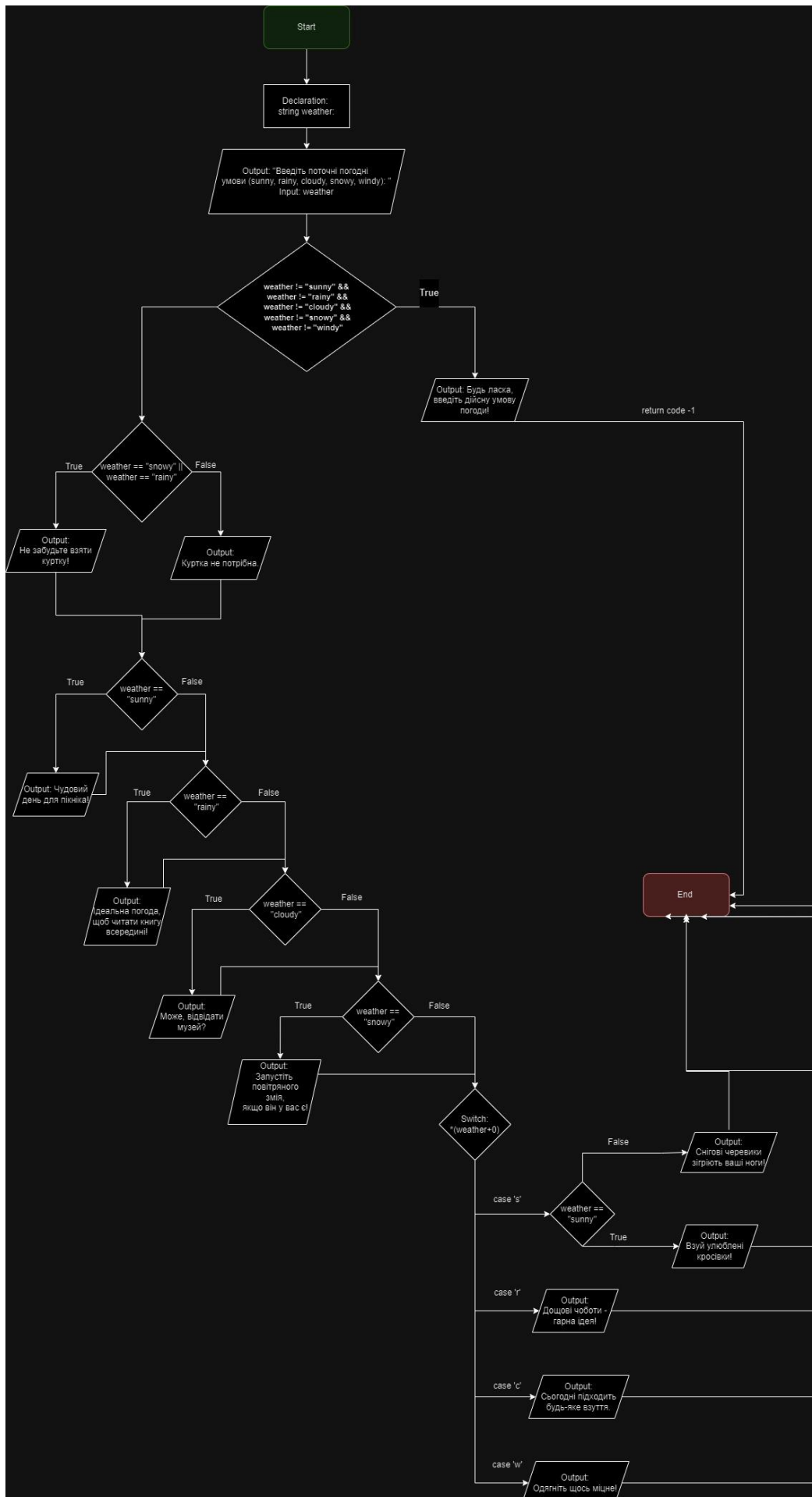
    else if (weather == "snowy") {
        cout << "Як щодо того, щоб зліпити сніговика?" << endl;
    }
    }
    else if (weather == "windy") {
        cout << "Запустіть повітряного змія, якщо він у вас є!" << endl;
    }
    }

    switch (weather[0]) {
    case 's':
        if (weather == "sunny") {
            cout << "Взуй улюблені кросівки!" << endl;
        }
        else {
            cout << "Снігові черевики зірпють ваші ноги!" << endl;
        }
        break;
    case 'r':
        cout << "Дощові чоботи - гарна ідея!" << endl;
        break;
    case 'c':
        cout << "Сьогодні підходить будь-яке взуття." << endl;
        break;
    case 'w':
        cout << "Одягніть щось міцне!" << endl;
        break;
    }

    return 0;
}

```

Flowchart Diagram:



Завдання № 7

Requirements :

Self Practice Task

Time:

Expected: 1 hour

Spent: 3-4 hours

```
✓ #include <iostream>
#include <string>
#include <algorithm>
using namespace std;

✓ int main() {
    string a, b;

    cin >> a >> b;

    string result;
    int carry = 0;
    int i = a.size() - 1;
    int j = b.size() - 1;

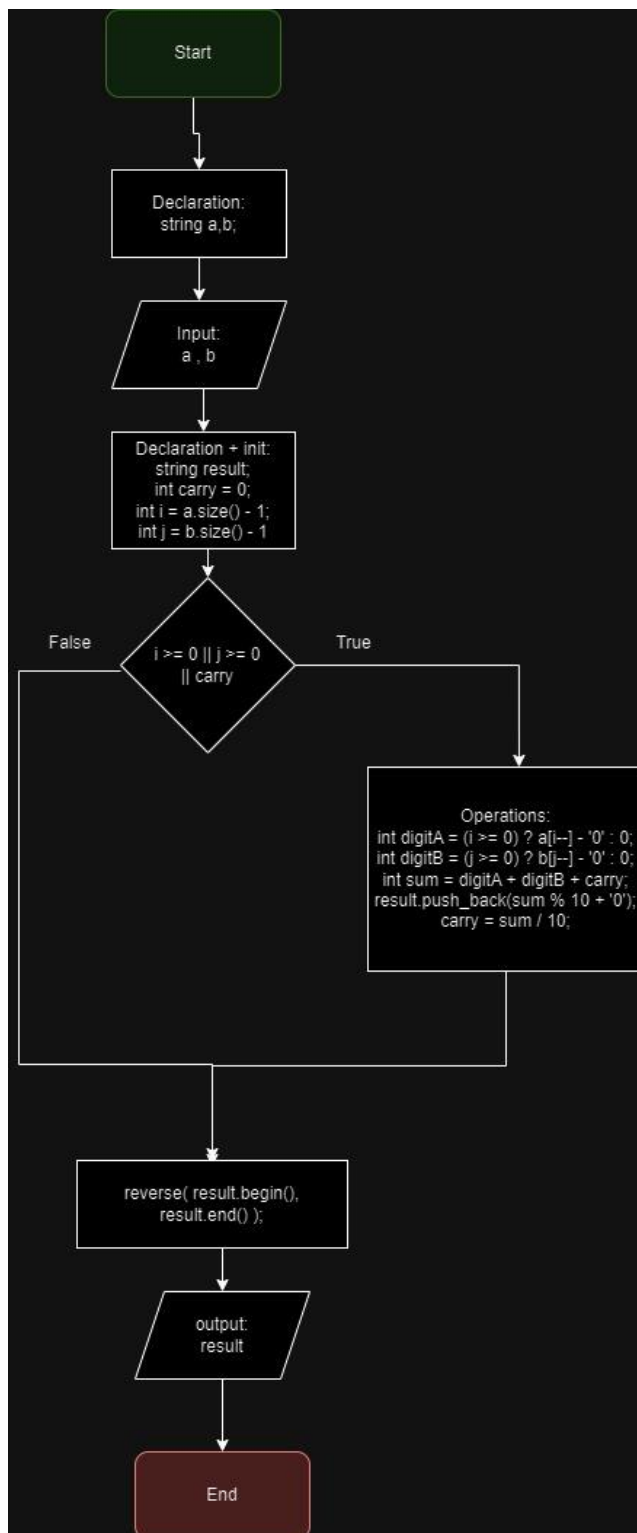
    while (i ≥ 0 || j ≥ 0 || carry) {
        int digitA = (i ≥ 0) ? a[i--] - '0' : 0;
        int digitB = (j ≥ 0) ? b[j--] - '0' : 0;
        int sum = digitA + digitB + carry;
        result.push_back(sum % 10 + '0');
        carry = sum / 10;
    }

    reverse(result.begin(), result.end());

    cout << result << endl;

    return 0;
}
```

Flowchart Diagram:



Created	Problem	Compiler	Result	Time (sec)	Memory (MB)	#
2 hours ago	1346 - C++	C++ 17	Accepted	0.008	1.492	1799504

Pull Request: [Here](#)

Висновок:

Я поглибив свої знання з побудовою діаграм в drawio . Також закріпив основні конструкції C та C++, пригадав особливості цієї мови програмування.