

Міністерство освіти і науки України  
Національний університет «Львівська політехніка»  
Кафедра систем штучного інтелекту



## **Звіт**

**про виконання розрахунково-графічних робіт блоку № 7**  
**з дисципліни:** «Основи програмування»

до:

ВНС Розрахунково-графічних робіт № 1-4  
Практичних Робіт до блоку № 7

**Виконав:**  
Студент групи ШІ-11  
Силіч Богдан

Львів 2024

## Мета роботи:

Закріпити вивчений матеріал на практиці, зокрема, набути навичок роботи з файлами як у мові C, так і в C++, а також удосконалити вміння працювати зі строками C++.

## Теоретичні відомості:

Теми пройдені під час семестру

## Джерела використані для опрацювання теоретичного матеріалу:

Ознайомився під час навчання

## Виконання роботи:

### Завдання №1 VNS Practice Work Task 1-20

**Варіант 20.**  $\alpha = \frac{a^x + b^{-x} \sin(a-b)}{\sqrt{|a-b|}}; \beta = a.e^{-\sqrt{a}} \cos(bx/a); a=0,5;$

$b=2,9; x=0,3.$

### Завдання №2 VNS Practice Work Task 2-6

**Варіант 6.**  $y = \begin{cases} \frac{x}{15x^2}, & x^2 + y^2 \leq a^2, \\ x^2 + e^x, & x^2 + y^2 > a^2; \end{cases}$

де  $x \in [-3,3]; h_x = 0,5; a = 5,3; y = 1,1.$

### Завдання №3 VNS Practice Work Task 3-1

**Варіант 1.** Обчислення об'єму паралелепіпеда. Нижче приведений вид екрану під час виконання програми, що рекомендується (дані, які вводяться користувачем, виділені напівжирним шрифтом).

Обчислення; об'єму паралелепіпеда.

Введіть початкові дані:

Довжина (см) > **9**

Ширина(см)> **7.5**

Висота(см)> **5**

Об'єм: 337.50 куб.см.

### Завдання №4 VNS Practice Work Task 4-1

**Варіант 1.** Знайти суму цифр введеного з клавіатури числа.

### Завдання №5 Algotester Task 0346

# Сума

Обмеження: 2 сек., 256 MiB

Обчислити суму двох цілих невід’ємних чисел.

## Вхідні дані

Два цілих числа  $a$  і  $b$  записаних в одному рядку.

## Вихідні дані

Виведіть суму  $a + b$ .

## Обмеження

$0 \leq a, b < 10^{111111}$ .

# Завдання №6 Algotester Task 1754

### Шифрування корупційних схем

Обмеження: 2 сек., 256 MiB

Задачі «Шифрування корупційних схем» і «Дешифрування корупційних схем» пов'язані між собою, але їх можна розв'язувати окремо одну від одної. Наступні чотири абзаци є однаковими в обох умовах.

В офісі Зеніка працює один татарин. Щодня він придумує нову геніальну корупційну схему й опісля ділиться нею із Зеніком у повідомленні. Цього місяця знову не обійшлося без нової хитрої схеми. Однак Татарин — не дурний. Він знає, що повідомлення зі схемою для Зеніка можуть перекопати правоохоронці. Тому він шифрує його алгоритмом кодування довжин серій.

Кодування довжин серій (англ. Run-length encoding, RLE) — простий алгоритм стиснення даних, який оперує серіями даних, тобто послідовностями, у яких один і той ж символ зустрічається кілька разів поспіль. При кодуванні рядок однакових символів, що становлять серію, замінюють рядком, який містить сам повторюваний символ і кількість його повторів (3 Вікіпедії).

Розглянемо приклад кодування рядка `AAAA88888888AC8888000000000000`. Якщо застосувати алгоритм RLE до нього, то отримаємо `4A781A1C4811D`. Останній запис інтерпретується як чотири `A`, сім `8`, одна `A`, одна `C`, чотири `8`, одинадцять `0`.

Зверніть увагу, що `4A48381A1C487D4D` не є правильно закодованим рядком — треба записувати цілі серії. `4A78AC4811D` також не є правильно закодованим рядком — навіть якщо довжина серії символів дорівнює одиниці, то все одно потрібно записувати `1`.

Зашифруйте корупційну схему татарина алгоритмом RLE.

#### Вхідні дані

Вхідні дані містять єдиний рядок  $s$  — корупційну схему татарина.

#### Вихідні дані

Виведіть закодовану за алгоритмом RLE корупційну схему.

Довжина закодованої схеми не перевищуватиме  $10^6$  символів.

#### Обмеження

$1 \leq |s| \leq 10^6$ ,

$s$  складається з великих латинських літер.

# Завдання №7 Algotester Task 0941

### Максимум

Обмеження: 2 сек., 256 MiB

Написати програму, яка знаходить значення найбільшого елемента масиву.

#### Вхідні дані

У першому рядку задано одне натуральне число  $n$  — розмір масиву.

У другому рядку задано  $n$  цілих чисел — елементи масиву.

#### Вихідні дані

У єдиному рядку виведіть одне ціле число — значення найбільшого елемента масиву.

#### Обмеження

$0 < n < 1000$ ,

$-12345678 \leq \text{елемент масиву} \leq 12345678$ .

#### Приклади

Вхідні дані (stdin)	вихідні дані (stdout)
3 17 10 5	17
Вхідні дані (stdin)	вихідні дані (stdout)
8 -10 34 -200 120 11 0 9 -1	120

# Завдання №8 Algotester Task 1561

## Промова президента

Обмеження: 2 сек., 256 МБ

Іноді президенту стає сумно, через те, що його ніхто не розуміє, тому у наступній промові він вирішив повторити  $n$  разів твердження  $x$ , з надією, що його хтось таки зрозуміє. Не так просто  $n$  разів повторити одне твердження без помилок. Вам потрібно сказати чи впорався президент із завданням.

### Вхідні дані

У першому рядку задане одне ціле число  $n$ .

У другому рядку задане твердження  $x$ .

У третьому рядку задані  $n$  тверджень через пробіл.

### Вихідні дані

Виведіть **YES**, якщо президент впорався із завданням, інакше виведіть **NO**.

### Обмеження

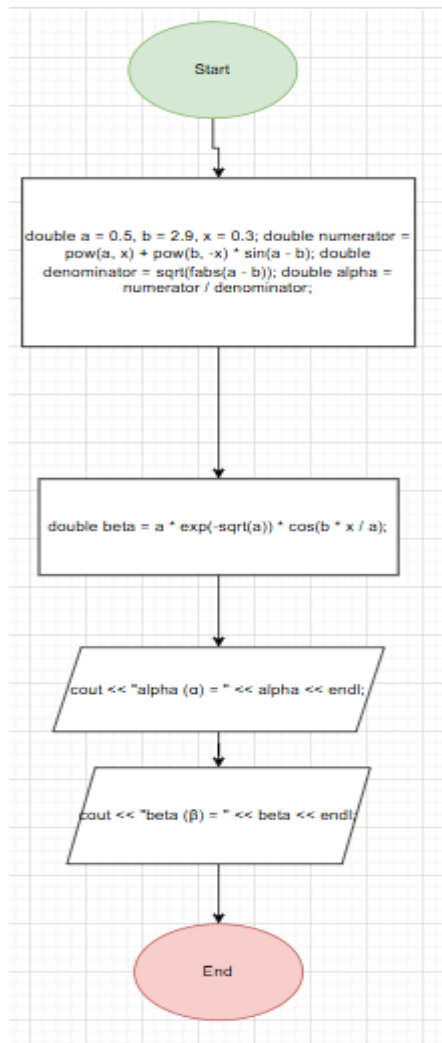
$1 \leq n \leq 10$ ,

довжина будь-якого твердження не перевищує 10,

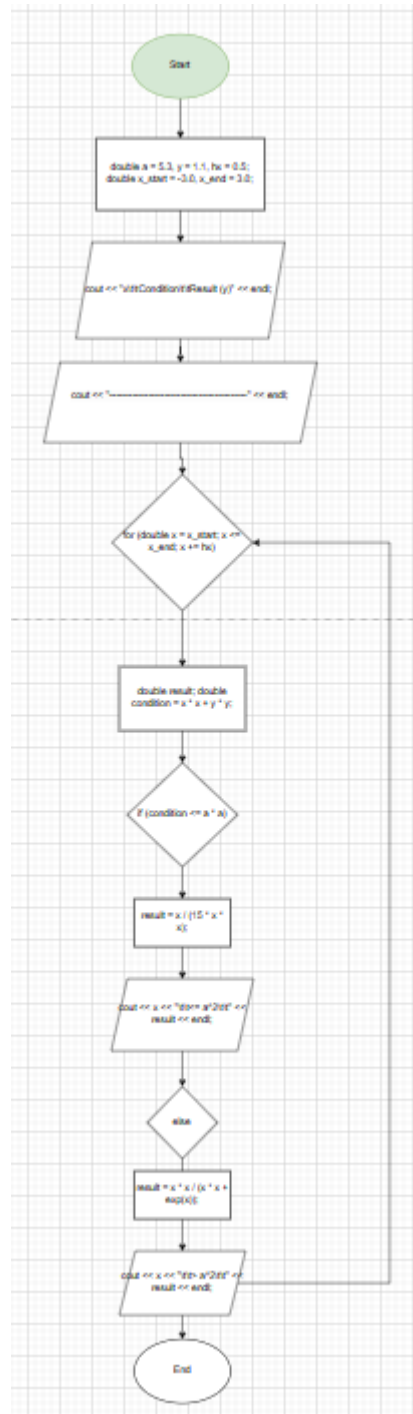
кожне твердження складається виключно з маленьких літер англійського алфавіту.

## Дизайн та планована оцінка часу виконання завдань

### Завдання №1 VNS Practice Work Task 1-20



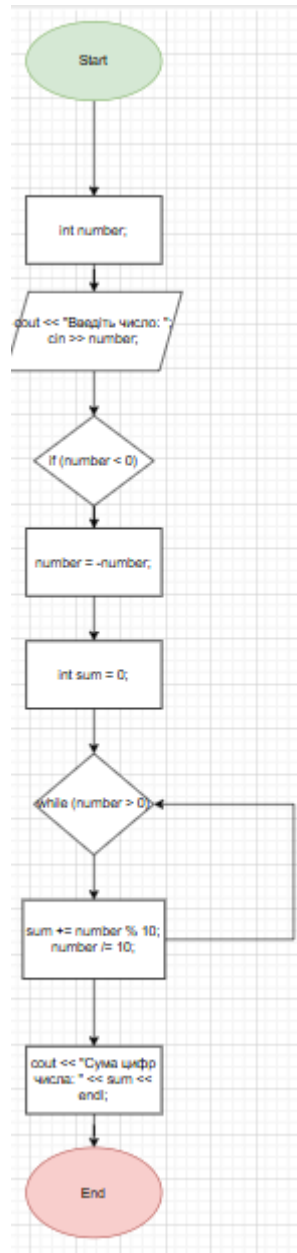
## Завдання №2 VNS Practice Work Task 2-6



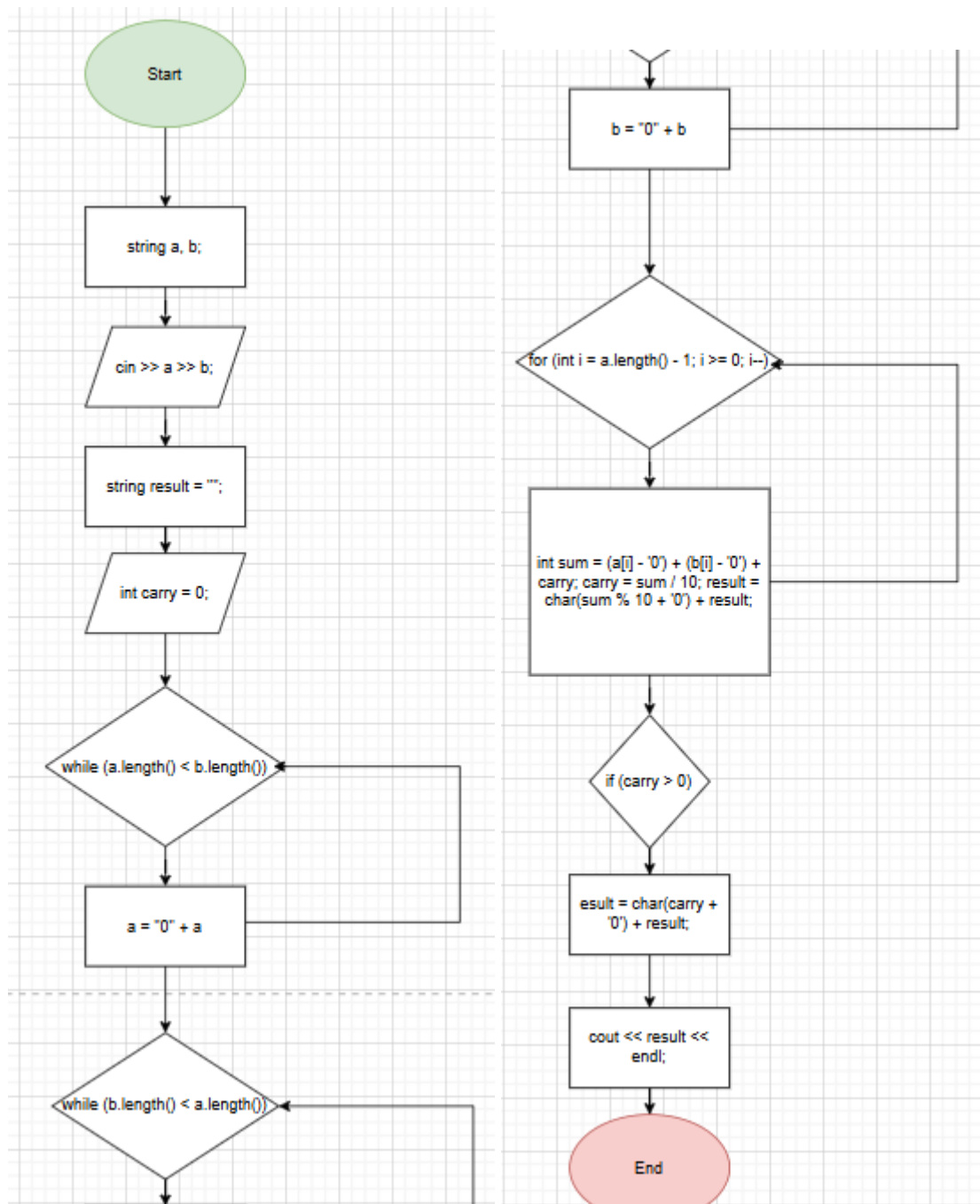
## Завдання №3 VNS Practice Work Task 3-1



## Завдання №4 VNS Practice Work Task 4-1

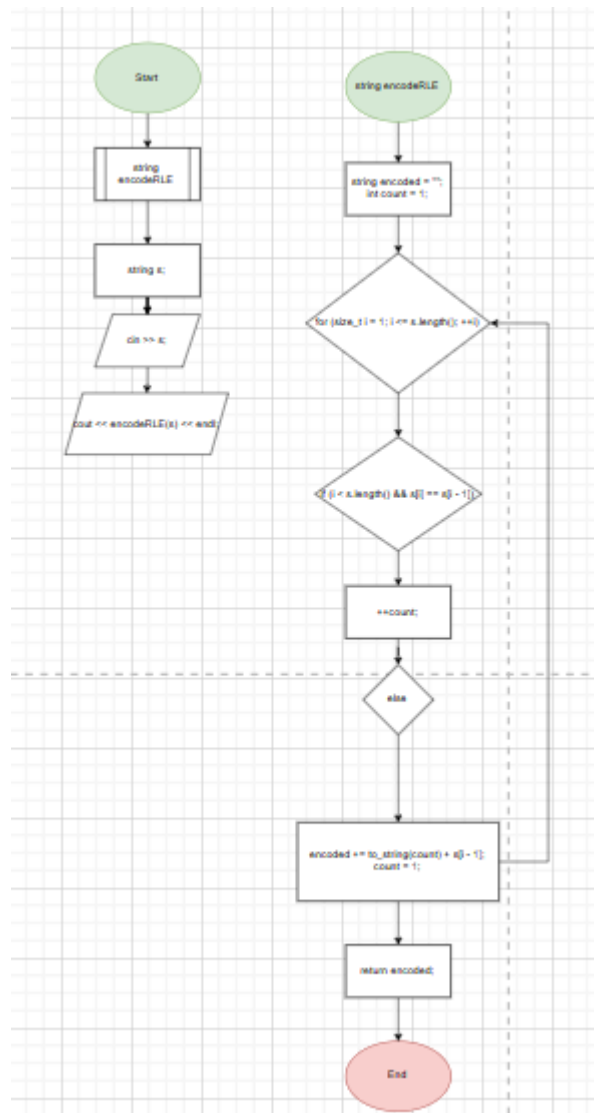


## Завдання №5 Algotester Task 0346

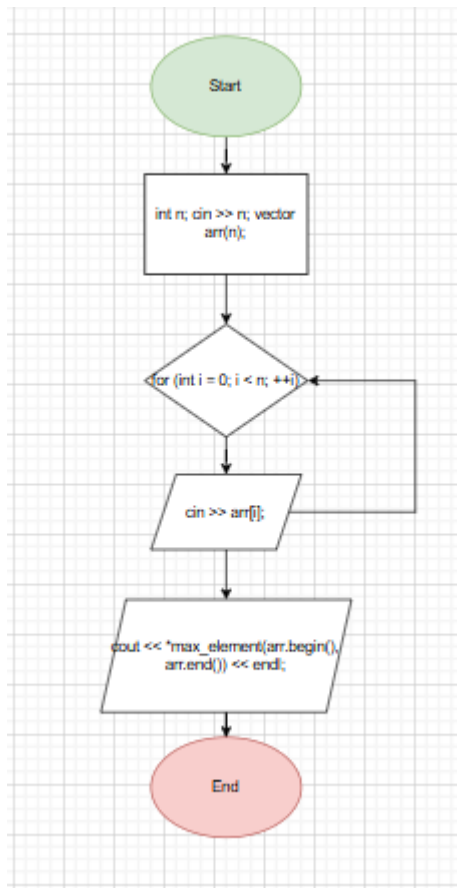


## Завдання №6 Algotester Task 1754

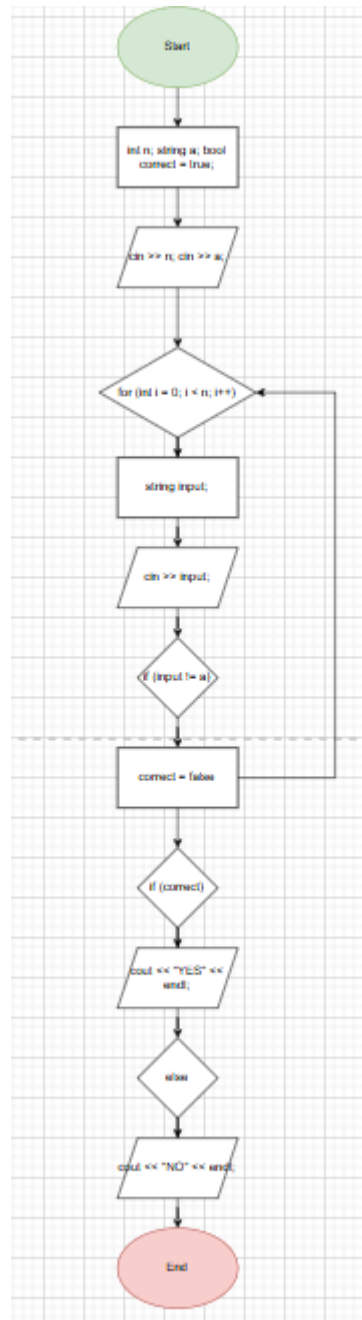




**Завдання №7 Algotester Task 0941**



## Завдання №8 Algotester Task 1561



## Код програм

### Завдання №1 VNS Practice Work Task 1-20

```
#include <iostream>
#include <cmath>

using namespace std;

int main() {
    double a = 0.5, b = 2.9, x = 0.3;
    double numerator = pow(a, x) + pow(b, -x) * sin(a - b);
    double denominator = sqrt(fabs(a - b));
    double alpha = numerator / denominator;

    double beta = a * exp(-sqrt(a)) * cos(b * x / a);

    cout << "alpha ( $\alpha$ ) = " << alpha << endl;
    cout << "beta ( $\beta$ ) = " << beta << endl;

    return 0;
}
```

```
alpha ( $\alpha$ ) = 0.207513
beta ( $\beta$ ) = -0.0415158
```

+ - 30 хв

### Завдання №2 VNS Practice Work Task 2-6

```
#include <iostream>
#include <cmath>

using namespace std;

int main() {
    double a = 5.3, y = 1.1, hx = 0.5;
    double x_start = -3.0, x_end = 3.0;

    cout << "x\t\tCondition\t\tResult (y)" << endl;
    cout << "-----" << endl;

    for (double x = x_start; x <= x_end; x += hx) {
        double result;
        double condition = x * x + y * y;

        if (condition <= a * a) {
            result = x / (15 * x * x);
            cout << x << "\t\t<= a^2\t\t" << result << endl;
        } else {
            result = x * x / (x * x + exp(x));
            cout << x << "\t\t> a^2\t\t" << result << endl;
        }
    }

    return 0;
}
```

-----		
-3	<= a^2	-0.0222222
-2.5	<= a^2	-0.0266667
-2	<= a^2	-0.0333333
-1.5	<= a^2	-0.0444444
-1	<= a^2	-0.0666667
-0.5	<= a^2	-0.133333
0	<= a^2	nan
0.5	<= a^2	0.133333
1	<= a^2	0.0666667
1.5	<= a^2	0.0444444
2	<= a^2	0.0333333
2.5	<= a^2	0.0266667
3	<= a^2	0.0222222

+ - 20 xB

## Завдання №3 VNS Practice Work Task 3-1

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

int main() {

    double length, width, height, weightPounds;

    cout << "Обчислення об'єму паралелепіпеда.\n";
    cout << "Введіть початкові дані:\n";

    cout << "Довжина (см) > ";
    cin >> length;

    cout << "Ширина (см) > ";
    cin >> width;

    cout << "Висота (см) > ";
    cin >> height;

    double volume = length * width * height;

    cout << fixed << setprecision(2);
    cout << "Об'єм: " << volume << " куб.см.\n";

    cout << "\nПерерахунок ваги:\n";
    cout << "Введіть вагу в фунтах > ";
    cin >> weightPounds;

    double weightKilograms = (weightPounds * 405.9) / 1000;

    cout << "Вага: " << weightKilograms << " кг.\n";

    return 0;
}
```

```
Обчислення об'єму паралелепіпеда.
Введіть початкові дані:
Довжина (см) > 10
Ширина (см) > 10
Висота (см) > 10
Об'єм: 1000.00 куб.см.
```

```
Перерахунок ваги:
Введіть вагу в фунтах > 10
Вага: 4.06 кг.
```

+50 хв

## Завдання №4 VNS Practice Work Task 4-1

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

int main() {
    int number;
    cout << "Введіть число: ";
    cin >> number;
    if (number < 0) {
        number = -number;
    }

    int sum = 0;
    while (number > 0) {
        sum += number % 10;
        number /= 10;
    }

    cout << "Сума цифр числа: " << sum << endl;

    return 0;
}
```

```
Введіть число: 23123
Сума цифр числа: 11
```

**+ - 15 хв**

## Завдання №5 Algotester Task 0346

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

int main() {
    string a, b;
    cin >> a >> b;
    string result = "";
    int carry = 0;

    while (a.length() < b.length()) a = "0" + a;
    while (b.length() < a.length()) b = "0" + b;

    for (int i = a.length() - 1; i >= 0; i--) {
        int sum = (a[i] - '0') + (b[i] - '0') + carry;
        carry = sum / 10;
        result = char(sum % 10 + '0') + result;
    }

    if (carry > 0) {
        result = char(carry + '0') + result;
    }

    cout << result << endl;

    return 0;
}
```

Створено	Компілятор	Результат	Час (сек.)	Память (MB)	Дії
грудень 2014	C++ 23	Зарезоновано	0.930	2.125	<a href="#">Перегляд</a>

**+ - 40 хв**



## Завдання №6 Algotester Task 1754

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

string encodeRLE(const string& s) {
    string encoded = "";
    int count = 1;

    for (size_t i = 1; i <= s.length(); ++i) {
        if (i < s.length() && s[i] == s[i - 1]) {
            ++count;
        } else {
            encoded += to_string(count) + s[i - 1];
            count = 1;
        }
    }

    return encoded;
}

int main() {
    string s;
    cin >> s;
    cout << encodeRLE(s) << endl;
    return 0;
}
```

Створено	Компілятор	Результат	Час (сек.)	Пам'ять (MB)	Дії
4 хвилини тому	C++ 23	Зарховано	0.005	1.180	Перегляд

**+30 хв**

## Завдання №7 Algotester Task 0941

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

int main() {
    int n;
    cin >> n;
    vector<int> arr(n);

    for (int i = 0; i < n; ++i) {
        cin >> arr[i];
    }

    cout << *max_element(arr.begin(), arr.end()) << endl;

    return 0;
}
```

Створено	Компілятор	Результат	Час (сек.)	Пам'ять (MB)	Дії
годину тому	C++ 23	Зарховано	0.002	1.098	Перегляд

**+20 хв**

## Завдання №8 Algotester Task 1561

```

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

int main() {
    int n;
    string a;
    bool correct = true;

    cin >> n;
    cin >> a;

    for (int i = 0; i < n; i++) {
        string input;
        cin >> input;
        if (input != a) {
            correct = false;
        }
    }

    if (correct) {
        cout << "YES" << endl;
    } else {
        cout << "NO" << endl;
    }

    return 0;
}

```

Створено	Компілятор	Результат	Час (сек.)	Пам'ять (MB)	Дії
депільна секунда тому	C++ 23	Зарезеровано	0.002	0.949	Перегляд

**+ - 30 хв**

## Висновок:

Виконуючи цю розрахункову роботу, я значно вдосконалив свої знання та практичні навички у мові програмування C++. Робота дозволила глибше зрозуміти основні принципи програмування, структури даних та алгоритми, а також розвинути здатність вирішувати більш складні завдання з використанням цього інструменту. Цей досвід допоможе мені у подальшому ефективніше застосовувати мову C++ для розв'язання різноманітних практичних задач.