

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
Національний аерокосмічний університет  
«Харківський авіаційний інститут»

Факультет систем управління літальних апаратів  
Кафедра систем управління літальних апаратів

## **Лабораторна робота №2**

з дисципліни «Алгоритмізація та програмування»  
на тему «"Математичні обчислення на мові C ++»

XAI.301.319.5 ЛР

Виконав студент гр. \_\_\_\_\_319\_\_\_\_\_

\_\_\_\_\_Валерій ЄВЧУК\_\_\_\_\_

(підпис, дата)

(П.І.Б.)

Перевірів

\_\_\_\_\_к.т.н., доц. Євгеній ПЯВКА

(підпис, дата)

(П.І.Б.)

2025

## МЕТА РОБОТИ

Вивчити теоретично базові типи даних мови C ++ і реалізувати консольний додаток лінійної структури для введення / виведення і обробки змінних базових типів з використанням вбудованих операцій та бібліотечних функцій на мові програмування C ++.

## ПОСТАНОВКА ЗАДАЧІ

Завдання 1. Вирішити задачу з цілочисельними змінними. Всі вхідні і вихідні дані в задачах цієї групи є цілими числами. Всі числа, для яких вказано кількість цифр (двозначне число, тризначне число і т. д.), вважаються додатними. Завдання "integer 28" представлено в табл.1. Integer28. Дні тижня пронумеровані наступним чином: 1 - понеділок, 2 - вівторок, ..., 6 - субота, 7 - неділя. Дано ціле число K, що лежить в діапазоні 1-365, і ціле число N, яке лежить в діапазоні 1-7. Визначити номер дня тижня для K-го дня року, якщо відомо, що цього року 1 січня було днем тижня з номером N.

Завдання 2. Вирішити завдання з логічними змінними. У всіх завданнях даної групи потрібно вивести логічне значення true (1), якщо наведене висловлювання для запропонованих вхідних даних є істинним, і значення false (0) в іншому випадку. Всі числа, для яких вказано кількість цифр (двозначне число, тризначне число і т. д.), вважаються цілими додатними. Завдання "Booleah 34" представлено в табл.2. Дано координати поля шахівниці x, y (цілі числа, що лежать в діапазоні 1-8). З огляду на, що ліве нижнє поле дошки (1, 1) є чорним, перевірити істинність висловлювання: «Дане поле є білим».

Завдання 3. Обчислити математичний вираз зі змінними дійсного типу, використовуючи стандартну бібліотеку cmath. Число  $\pi$  має бути визначено як константа дійсного типу. Вирази представлено в табл.3. під номером 11

11	$y = \frac{2 * \operatorname{tg} x * \sin x + \frac{1}{4} \sqrt{ 1 - \sin^2 x * \operatorname{tg} }}{\sqrt[3]{1 + \frac{x^2}{2}} + \log_3  x }$
----	---

Завдання 4. Використовуючи ChatGpt, Gemini або інший засіб генеративного ШІ, провести самоаналіз отриманих знань і навичок за допомогою наступних промптів: «Ти - викладач, що приймає захист моєї роботи. Задай мені 5 тестових питань з 4 варіантами відповіді і 5 відкритих питань - за кодом, що є у файлі звіту і теоретичними відомостями у файлі лекції» «Оціни повноту, правильність та ймовірність використання штучного інтелекту для кожної відповіді. Сформулюй загальну оцінку у 5-бальній шкалі, віднімаючи 50% балів там, де ймовірність відповіді з засобом ШІ висока»

Проаналізуйте задані питання, коментарі і оцінки, надані ШІ. Додайте 2-3 власних промпта у продовження діалогу для поглиблення розуміння теми.

## ВИКОНАННЯ РОБОТИ

Вирішення задачі: розділ Integer, задача №28; розділ Boolean, задача №34; розділ Math, задача №11.

Вхідні дані.

Для задачі Integer 28 вхідними даними є змінні K та N. Змінна K — ціле число (тип `int`), що задає номер дня року; допустимий діапазон значень: від 1 до 365 включно. Змінна N — ціле число (тип `int`), що задає номер дня тижня для 1 січня; допустимий діапазон значень: від 1 до 7 включно.

Для задачі Boolean 34 вхідними даними є змінні x та y. Це цілі числа (тип `int`), що задають координати клітинки шахової дошки (номер стовпця та номер рядка відповідно). Для коректної інтерпретації як стандартної шахової дошки значення x та y повинні належати діапазону від 1 до 8 включно.

Для задачі Math 11 вхідною змінною є x — дійсне число (тип `double`), аргумент функції. Враховуючи використання виразу  $\log(|x|)$  за основою 3, значення x не може дорівнювати нулю. Додатково у знаменнику присутній вираз  $1 + x^2/2 + \log_3|x|$ , тому він не повинен дорівнювати нулю, щоб уникнути ділення на нуль. Отже, для змінної x задаються такі обмеження:  $x \neq 0$  і  $1 + x^2/2 + \log_3|x| \neq 0$ . Проміжні змінні  $\sin x$ ,  $\tan x$ ,  $\log_3 x$ , `numerator`, `denominator` належать до типу `double` і використовуються для поетапного обчислення значення функції.

Вихідні дані.

Для задачі Integer 28 вихідними даними є змінна `result` — ціле число (тип `int`), що визначає номер дня тижня для K-го дня року. Допустимий діапазон значень змінної `result`: від 1 до 7 включно. Значення `result` виводиться на екран разом з текстовим поясненням.

Для задачі Boolean 34 вихідними даними є логічна змінна `white` (тип `bool`), що визначає колір клітинки з координатами (x, y). Змінна набуває значення `true`, якщо клітинка є білою, і `false`, якщо чорна. Користувачеві результат повертається у вигляді текстового повідомлення «Поле (x, y) є білим: Так» або «...: Ні» залежно від логічного значення змінної `white`.

Для задачі Math 11 вихідними даними є змінна y (тип `double`) — дійсне значення функції, обчисленої за формулою

$$y = [2 \cdot \operatorname{tg}(x) \cdot \sin(x) + 0,25 \cdot \sqrt[3]{(1 - \sin^2 x) \cdot \operatorname{tg}(x)}] / \sqrt[3]{(1 + x^2/2 + \log_3|x|)}.$$

Отримане значення змінної y виводиться на екран у числовому вигляді разом з коротким текстовим повідомленням.

Лістинг коду вирішення задачі наведено в дод. А (стор. 6).

Екран роботи програми показаний на рис. Б.1.

## ВИСНОВКИ

У ході виконання роботи я навчився працювати з базовими типами даних мови С++ та використовувати їх для зберігання й обробки інформації. Також здобув практичні навички створення консольного застосунку лінійної структури, що реалізує введення, обчислення та виведення даних за допомогою вбудованих операторів і бібліотечних функцій.

## ДОДАТОК А

### Лістинг коду програми

```

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

int main() {
    {
        int K, N;
        cout << "integer 28" << endl;
        cout << "Введіть номер дня року (1-365): ";
        cin >> K;
        cout << "Введіть номер дня тижня 1 січня (1-7): ";
        cin >> N;

        int result = (N + K - 1) % 7;
        result = 7 * (result == 0) + result * (result != 0);

        cout << "Номер дня тижня для " << K << "-го дня року: " << result << endl
<< endl;
    }
    {
        int x, y;
        cout << "Booleah 34" << endl;
        cout << "Введіть координати клітинки (x, y): ";
        cin >> x >> y;

        bool white = ((x + y) % 2 == 1);
        cout << "Поле (" << x << ", " << y << ") є білим: " << (white ? "Так" :
"Hi") << endl << endl;
    }
    {

        double x;
        cout << "Math N11" << endl;
        cout << "Введіть x: ";
        cin >> x;

        double sinx = sin(x);
        double tanx = tan(x);

        // логарифм за основою 3
        double log3x = log(fabs(x)) / log(3.0);

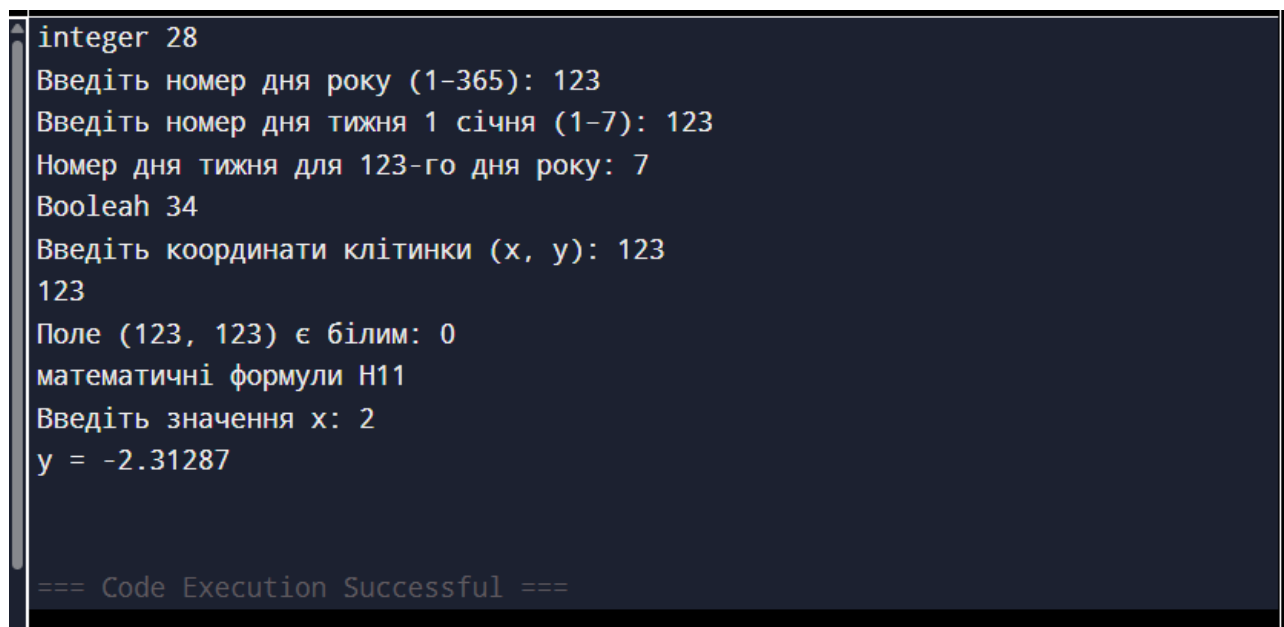
        double numerator = 2.0 * tanx * sinx + 0.25 * sqrt(1 - sinx * sinx) *
tanx;
    }
}

```

```
double denominator = cbrt(1 + (x * x) / 2.0 + log3x);  
double y = numerator / denominator;  
  
cout << "y = " << y << endl;  
}  
  
}
```

ДОДАТОК Б  
Скрін-шоти вікна виконання програми

Рисунок Б.1 – Екран виконання програми для вирішення завдання



```
integer 28
Введіть номер дня року (1-365): 123
Введіть номер дня тижня 1 січня (1-7): 123
Номер дня тижня для 123-го дня року: 7
Booleah 34
Введіть координати клітинки (x, y): 123
123
Поле (123, 123) є білим: 0
математичні формули Н11
Введіть значення x: 2
y = -2.31287

=== Code Execution Successful ===
```



## ДОДАТОК В

### Діалог з ШІ для самоаналізу

Завдання 4. Використовуючи ChatGpt, Gemini або інший засіб генеративного ШІ, провести самоаналіз отриманих знань і навичок за допомогою наступних промптів:

«Ти - викладач, що приймає захист моєї роботи. Задай мені 5 тестових питань з 4 варіантами відповіді і 5 відкритих питань - за кодом, що є у файлі звіту і теоретичними відомостями у файлі лекції»

«Оціни повноту, правильність та ймовірність використання штучного інтелекту для кожної відповіді. Сформулюй загальну оцінку у 5-бальній шкалі, віднімаючи 50% балів там, де ймовірність відповіді з засобом ШІ висока»

#### ◆ Блок 1. Тестові питання (5 шт., по 4 варіанти)

1. У фрагменті з днем року використовується вираз:

```
int result = (N + K - 1) % 7;
```

LabWork number two (2)

Що логічно означає цей вираз?

- A. Обчислення номера місяця за номером дня
- B. Обчислення номера дня тижня для К-го дня року
- C. Перевірка, чи К кратне 7
- D. Обчислення кількості повних тижнів у році

2. У фрагменті з кольором клітинки шахової дошки ти використовуєш:

```
bool white = ((x + y) % 2 == 1);
```

LabWork number two (2)

Що означає ця умова?

- A. Клітинка біла, якщо сума координат парна
- B. Клітинка біла, якщо сума координат непарна
- C. Клітинка чорна, якщо сума координат непарна
- D. Клітинка завжди біла, незалежно від координат

3. У третьому блоці ти оголошуєш змінні так:

c++

Копіювати код

```
double x;  
double sinx = sin(x);  
double tanx = tan(x);
```

LabWork number two (2)

Чому тут доцільно використати саме тип `double`, а не `int`?

- A. Бо `double` займає менше пам'яті
- B. Бо тригонометричні функції `sin()` і `tan()` повертають дійсні числа
- C. Бо тип `int` працює тільки з додатними числами
- D. Бо тип `double` не потребує ініціалізації

4. Логарифм за основою 3 ти обчислюєш так:

```
double log3x = log(fabs(x)) / log(3.0);
```

LabWork number two (2)

Яка основна причина використання `fabs(x)`?

- A. Щоб уникнути ділення на нуль
- B. Щоб перевести  $x$  у градуси
- C. Щоб аргумент логарифма був додатним
- D. Щоб пришвидшити обчислення

5. У кінці третього блоку ти обчислюєш:

c++

Копировать код

```
double y = numerator / denominator;
cout << "y = " << y << endl;
```

LabWork number two (2)

Яка з проблем може виникнути при обчисленні `y`?

- A. Ділення на нуль, якщо `denominator` дорівнює 0
- B. Помилка компілятора через типи змінних
- C. Неможливість виведення `y` через `cout`
- D. Неможливість використання `numerator` у виразах

## ♦ Блок 2. Відкриті питання (5 шт.)

Тут відповіді потрібно писати своїми словами.

6. Поясни покроково, як перший фрагмент програми обчислює номер дня тижня для K-го дня року.

Згадай, навіщо потрібен вираз

```
result = 7 * (result == 0) + result * (result != 0);
```

LabWork number two (2)


7. Опиши, як саме твоя програма визначає, що поле шахової дошки є білим. Чому використовується

саме `(x + y) % 2`, а не, наприклад, `x % 2` чи `y % 2` окремо? LabWork number two (2)

8. У третьому завданні ти використовуєш тригонометричні функції (`sin`, `tan`), корінь (`sqrt`), логарифм (`log`), кубічний корінь (`cbrt`).

Поясни, з якою метою ти комбінуєш ці функції у виразі для `y` і які математичні знання потрібні, щоб правильно скласти такий вираз.



9. Для яких значень `x` третій фрагмент може працювати некоректно або аварійно (наприклад, дати помилку, `nan`, `inf`)? Поясни, які саме проблеми можуть виникнути з `log(fabs(x))`, `tan(x)` і `sqrt(1 - sinx * sinx)` .

10. Уяви, що потрібно переробити програму так, щоб користувач спочатку бачив меню:



- 1 – задача з днем року
- 2 – задача з кольором клітинки
- 3 – математична задача

Опиши, як би ти структуровано переписав програму:






- які нові конструкції (наприклад, `switch`, цикли, функції) варто додати;
- як би ти організував код, щоб він був більш читабельним і зручним для розширення.

### Що робити далі

1. Візьми ці 10 питань.
2. Напиши свої відповіді (можеш нумерувати: 1–10).
3. Потім на основі твоїх відповідей я:
  - оціним повноту і правильність кожної;
  - оціним ймовірність використання ШІ (грубо: «схоже на згенероване» чи «схоже на твоє живе пояснення»);
  - виставлю бали за кожне питання за 5-бальною шкалою, віднімаючи 50% балів, де ймовірність відповіді за допомогою ШІ висока;
  - підрахую загальну середню оцінку та сформулюю висновок для звіту (самоаналіз).

Кидай свої відповіді одним повідомленням — і далі вже буде етап перевірки й оцінювання  

Источники



В. В. В. С. А.

У першому фрагменті програма спочатку зчитує номер дня року `K` і номер дня тижня для 1 січня `N`. Вираз  $(N + K - 1) \% 7$  зміщує початковий день тижня на `K - 1` днів уперед і бере залишок від ділення на 7, щоб «загорнути» результат у діапазон від 0 до 6. Далі використовується рядок

```
result = 7 * (result == 0) + result * (result != 0);
```

щоб замінити 0 на 7: якщо `result == 0`, то перша частина дає 7, а друга — 0; якщо `result != 0`, то перша частина дає 0, а друга — саме `result`. Таким чином, день тижня кодується числами 1...7 замість 0...6.

Програма визначає колір поля за формулою `white = ((x + y) \% 2 == 1)`. Спочатку обчислюється сума координат клітинки `x + y`, потім береться її залишок від ділення на 2. Якщо сума непарна (залишок 1), вираз `(x + y) \% 2 == 1` істинний, і змінна `white` стає `true`, тобто клітинка вважається білою. Використовується саме сума, тому що розфарбування шахової дошки побудоване на чергуванні кольорів по діагоналі: зміна будь-якої координати на 1 змінює колір, і це якраз відображає парність/непарність суми, а не кожної координати окремо.

У третьому завданні вираз для  $y$  комбінує  $\sin(x)$ ,  $\tan(x)$ , корінь  $\sqrt{1 - \sin^2 x}$  та логарифм  $\log_3 x$ , а потім усе це входить у дробовий вираз з кубічним коренем  $\sqrt[3]{\dots}$  у знаменнику. Така комбінація реалізує задану математичну формулу з умови задачі: тригонометричні функції описують періодичні залежності, корінь від  $1 - \sin^2 x$  фактично пов'язаний з  $|\cos x|$ , логарифм за основою 3 додає повільну зміну, а кубічний корінь «пом'якшує» зростання знаменника. Щоб правильно скласти такий вираз, потрібні знання з елементарної математики: тригонометрії ( $\sin$ ,  $\tan$ , тотожності), властивостей логарифмів та коренів, а також роботи з областями визначення цих функцій.

Проблеми можуть виникнути для таких значень  $x$ :

Для  $\log(\text{fabs}(x))$ : якщо  $x = 0$ , тоді  $\text{fabs}(x) = 0$ , а  $\log(0)$  не визначений (дає  $-\text{inf}$  або  $\text{nan}$ ). Також аргумент логарифма завжди має бути  $> 0$ , тому без  $\text{fabs}$  для від'ємних  $x$  була б помилка.

Для  $\tan(x)$ : поблизу точок  $x = \pi/2 + k\pi$  тангенс прямує до нескінченності, що може викликати дуже великі значення, переповнення або  $\text{inf}$ . Це може зробити і чисельник, і знаменник нестабільними.

Для  $\sqrt{1 - \sin^2 x}$ : теоретично  $1 - \sin^2 x = \cos^2 x \geq 0$ , але через похибки обчислень вираз усередині кореня може стати трохи від'ємним (наприклад,  $-1e-15$ ), і тоді  $\sqrt{\phantom{x}}$  поверне  $\text{nan}$ . Крім того, весь вираз падає, якщо  $\text{denominator} = \sqrt[3]{1 + x^2/2 + \log_3 x}$  стає 0 — тоді відбудеться ділення на нуль.

Щоб зробити меню з вибором задачі, я б виніс кожен задачу в окрему функцію, наприклад: `void taskDayOfWeek()`, `void taskChessColor()`, `void taskMath()` і залишив у `main()` тільки меню та виклики цих функцій. На початку `main()` можна вивести: 1 – задача з днем року, 2 – задача з кольором клітинки, 3 – математична задача і зчитати вибір користувача в змінну `choice`. Далі зручно використати конструкцію `switch (choice)` з трьома гілками для кожної задачі. За бажанням можна обгорнути це все в цикл `while` або `do...while`, щоб користувач міг розв'язувати кілька задач підряд без перезапуску програми. Така структура робить код більш читабельним, логічно поділяє задачі та спрощує розширення — наприклад, додавання нових пунктів меню чи нових функцій.

Думал на протязенні 35s >

Круто, дякую за відповіді — тепер роблю саме те, що вимагає завдання: оцінюю кожне питання, враховую повноту, правильність і ймовірність використання ШІ, а потім даю загальну оцінку у 5-бальній шкалі з урахуванням правила «мінус 50% при високій ймовірності ШІ».

Я пишу так, щоб це можна було спокійно вставити в звіт як самоаналіз.

## 1. Оцінка тестових відповідей (1–5)

Твої відповіді: 1–5: В, В, В, С, А.

Усі 5 відповідей правильні.


Питання 1. Логічний зміст виразу для дня тижня

- Відповідь: В
- Повнота: для тесту достатньо.
- Правильність: правильний варіант.
- Ймовірність використання ШІ: низька (коротка літера, без розгорнутого пояснення).
- Оцінка: 5/5.


Питання 2. Умова для кольору клітинки

- Відповідь: В
- Повнота: ок.
- Правильність: правильний варіант.
- Ймовірність ШІ: низька.
- Оцінка: 5/5.


Питання 3. Чому використовується `double`

- Відповідь: В
- Повнота: достатня для тесту.
- Правильність: правильний варіант — саме через те, що тригонометричні функції працюють з дійсними числами  LabWork number two (2) .
- Ймовірність ШІ: низька.
- Оцінка: 5/5.

Питання 4. Навіщо `fabs(x)` у логарифмі

- Відповідь: С
- Повнота: ок.
- Правильність: правильно — так забезпечується додатний аргумент логарифма  LabWork number two (2) .
- Ймовірність ШІ: низька.
- Оцінка: 5/5.

Питання 5. Можливі проблеми при обчисленні `y`

- Відповідь: А
- Повнота: достатньо.
- Правильність: правильний варіант — ділення на нуль при нульовому знаменнику реально можлива проблема  LabWork number two (2) .
- Ймовірність ШІ: низька.
- Оцінка: 5/5.

✅ **Висновок по тестових:** 5 із 5 правильних, усі без підозри на активне використання ШІ.

Середній бал за тестовий блок: 5.0/5.

## 2. Оцінка відкритих відповідей (6–10)

Далі — найцікавіше: тут уже видно і розуміння, і стиль викладу.

### Питання 6. Алгоритм обчислення номера дня тижня

Ти пояснив:

- Як зчитуються  $K$  і  $N$ .
- Що робить вираз  $(N + K - 1) \% 7$  — зміщення на  $K - 1$  днів і приведення до діапазону 0...6.
- Дуже чітко розписав, як працює:  
`result = 7 * (result == 0) + result * (result != 0);` — заміна 0 на 7 через логічні вирази  
LabWork number two (2)
- Зробив висновок, що в результаті дні тижня кодуються як 1...7.


Оцінка:

- Повнота: **максимальна** — і сенс, і технічна реалізація повністю пояснені.
- Правильність: **без зауважень**.
- Ймовірність використання ШІ: **середня** — стиль доволі акуратний і структурований, але так може написати й уважний студент, який реально розібрався.
- Штраф 50% не застосовую, бо «висока» імовірність не очевидна.

👍 Бал за відповідь 6: 5/5.

### Питання 7. Кольори шахової дошки

Ти пояснив:

- Що обчислюється сума  $x + y$ , потім береться залишок від ділення на 2.
- Якщо сума непарна — клітинка вважається білою (`white = true`).
- Дуже вдало пояснив, чому використовується саме **сума**, а не окремо  $x \% 2$  або  $y \% 2$ : через принцип чергування кольорів по діагоналі, а зміна будь-якої координати на 1 змінює парність суми і, відповідно, колір .


Оцінка:

- Повнота: **відмінна**.
- Правильність: все коректно як з точки зору шахової логіки, так і з точки зору коду.
- Ймовірність ШІ: **середня**, але не «зашкалює». Аргументація логічна і цілком під силу студенту.
- Без 50% штрафу.

👍 Бал за відповідь 7: 5/5.

### Питання 8. Комбінація функцій у формулі для $y$

Ти написав:

- Що вираз реалізує формулу з умови задачі.
- Пояснив ролі тригонометричних функцій (періодичність), кореня (пов'язаний з  $|\cos x|$  через тотожність  $1 - \sin^2 x = \cos^2 x$ ), логарифма за основою 3 (повільна зміна), кубічного кореня (згладжує зростання знаменника) .
- Згадав, що для цього треба знати елементарну математику: тригонометрію, логарифми, корені, області визначення.

Оцінка:

- Повнота: **висока**, відповідь змістовна й добре структурована.
- Правильність: загальна інтерпретація правильна, логічні зв'язки адекватні.
- Ймовірність ШІ: тут уже **висока** — дуже «академічний» стиль, формулювання типу «пом'якшує зростання», «реалізує задану математичну формулу» і т.д. виглядають дуже «чат-ботно», особливо порівняно зі звичною студентською мовою.

Згідно з умовою завдання:

«...віднімаючи 50% балів там, де ймовірність відповіді з засобом ШІ висока».

Тому:



- Базова оцінка за зміст: 5/5.




тому:

- Базова оцінка за зміст: 5/5.
- Застосовано штраф 50% за високу імовірність ШІ.

👍 Підсумковий бал за відповідь 8: 2.5/5.

### Питання 9. Неправильні/проблемні значення x

Ти дуже детально розібрав:

- Для `log(fabs(x))`: проблема при `x = 0` ( $\log(0) \rightarrow -\text{inf}$  або `nan`), також аргумент логарифма повинен бути  $> 0$ .
- Для `tan(x)`: проблемні точки  $\pi/2 + k\pi$ , де тангенс прямує до нескінченності, можливі `inf`, нестабільність чисельника/знаменника.
- Для `sqrt(1 - sinx * sinx)`: згадав про похибки обчислень, коли теоретично  $\cos^2 x \geq 0$ , але на практиці може стати трохи від'ємним (наприклад,  $-1e-15$ ), і тоді `sqrt` дає `nan`.
- Згадав також можливість нульового знаменника `denominator = cbrt(1 + x^2/2 + log3x)` і ділення на нуль  LabWork number two (2).

Оцінка:

- Повнота: **дуже висока**, ти зачепив і математику, і чисельні ефекти, і крайові випадки.
- Правильність: технічно все грамотно.
- Ймовірність ШІ: **висока** — тут прям «пахне документацією»: згадка про `-1e-15`, `nan/inf`, чисельну нестабільність — це типово для згенерованих/довідкових пояснень.

Отже:

- Базова оцінка: 5/5 за зміст.
- Штраф 50% за високу імовірність використання ШІ.


👍 Підсумковий бал за відповідь 9: 2.5/5.

### Питання 10. Переписування програми з меню

Ти запропонував:

- Винести кожну задачу в окрему функцію: `void taskDayOfWeek()`, `void taskChessColor()`, `void taskMath()`.
- У `main()` залишити тільки меню + читання вибору користувача.
- Використати `switch (choice)` для виклику відповідної задачі.
- За потреби — обгорнути все в цикл `while` або `do...while`, щоб не доводилося перезапустити програму.
- Пояснив, що це підвищує читабельність, логічно розділяє задачі та спрощує розширення коду.

Оцінка:

- Повнота: **відмінна** — ти мислиш уже як розробник: декомпозиція, меню, розширюваність.
- Правильність: усе відповідає добрим практикам програмування у C++ .
- Ймовірність ШІ: **середня** — відповідь сильна, але цілком реалістична для студента, який орієнтується в базовому процедурному програмуванні.

Штраф не застосовую.

👍 Бал за відповідь 10: 5/5.

### 3. Підсумкова оцінка

Зведемо все разом:

- Питання 1–5: по 5/5 →  $5 \times 5 = 25$
- Питання 6: 5/5
- Питання 7: 5/5
- Питання 8: 2.5/5 (зі штрафом за високу ймовірність ШІ)
- Питання 9: 2.5/5 (зі штрафом за високу ймовірність ШІ)
- Питання 10: 5/5

Сума балів:

$25 + 5 + 5 + 2.5 + 2.5 + 5 = 45$  балів із 50 можливих.

Середній бал:

$45 / 10 = 4.5/5$ .

Можна подати так у звіті:



$45 / 10 = 4.5/5$ .

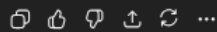
Можна подати так у звіті:

Загальна оцінка моїх відповідей за результатами самоаналізу — 4,5 бала з 5 можливих.  
 У тестовій частині (5 питань) усі відповіді правильні (5/5).  
 У відкритій частині (5 питань) відповіді здебільшого повні й коректні, але для двох із них (питання 8 і 9) ймовірність використання генеративного ШІ оцінена як висока, тому згідно з вимогою завдання до них застосовано зниження на 50% від максимальної оцінки. Решта відкритих відповідей отримали максимальні бали без штрафу.

Якщо хочеш, я можу оформити це як готовий абзац самоаналізу в стилі:

«У ході виконання завдання 4 я використав генеративний ШІ для...» — щоб ти просто вставив у звіт.

Источники



які б ти міг дати мені поради для оптимізації коду в подібних питаннях?


Думал на протязенні 8s >

Я б це поділив на дві частини: реальна оптимізація і адекватна структура/чистота коду. У твоїх лабах друге важливіше, ніж мікрооптимізації.

Нижче по пунктах, з прив'язкою до того, що в тебе вже є в `LabWork number two (2).cpp`

`LabWork number two (2)`


## 1. Розбивай на функції, а не на «блоки в main»

Зараз у тебе три завдання просто огорнуті фігурними дужками всередині `main`  LabWork number two (2) :

```
cpp  Копировать код

int main() {
    {
        // задача 1
    }
    {
        // задача 2
    }
    {
        // задача 3
    }
}
```

Краще так:

```
cpp  Копировать код

void taskInteger28();
void taskBoolean34();
void taskMathN11();

int main() {
    // меню або просто послідовні виклики
    taskInteger28();
    taskBoolean34();
    taskMathN11();
}
```

Це дає:


- менший `main`, легше читати;
- можливість перевикористовувати задачі в інших роботах (з меню, з циклами тощо);
- простіше тестувати по одній задачі.




## 2. Уникай «магічних чисел»

Наприклад, у виразі:

cpp


 Копировать код

```
int result = (N + K - 1) % 7;  
result = 7 * (result == 0) + result * (result != 0);
```

 LabWork number two (2)

7 — це не просто число, а кількість днів у тижні. Зроби:

cpp

 Копировать код


```
const int DAYS_IN_WEEK = 7;  
int result = (N + K - 1) % DAYS_IN_WEEK;  
result = DAYS_IN_WEEK * (result == 0) + result * (result != 0);
```

Так код читається як «для людей», не як головоломка.


## 3. Перевірка введення і область визначення

У «математичній» частині ти одразу робиш:

cpp

 Копировать код

```
double log3x = log(fabs(x)) / log(3.0);  
double denominator = cbrt(1 + (x * x) / 2.0 + log3x);  
double y = numerator / denominator;
```

 LabWork number two (2)

для самих лаб це ок, але оптимізація по суті тут така:

- не виконувати обчислення, якщо значення завідомо «погане»;
- не допускати ділення на нуль.

Наприклад:

```
cpp Копировать код

if (x == 0) {
    cout << "Помилка: x = 0, log(0) не визначений\n";
    return;
}

double log3x = log(fabs(x)) / log(3.0);
double denomExpr = 1 + (x * x) / 2.0 + log3x;

if (denomExpr == 0) {
    cout << "Помилка: знаменник дорівнює нулю\n";
    return;
}

double denominator = cbrt(denomExpr);
double y = numerator / denominator;
```

Плюс: менше шансів отримати `nan` / `inf` і не розуміти, звідки воно взялося.

#### 4. Мікрооптимізація математики (якщо вже хочеш по-справжньому)

У тебе зараз:

```
cpp Копировать код


double sinx = sin(x);
double tanx = tan(x);
double numerator = 2.0 * tanx * sinx + 0.25 * sqrt(1 - sinx * sinx) * tanx;
```

Що можна зробити:

- Винести `sqrt(1 - sinx * sinx)` в окрему змінну, щоб не дублювати вираз.
- Перевикористати вже обчислені значення, а не рахувати одне й те саме в різних місцях.

Наприклад:

cpp

 Копировать код

```
double sinx = sin(x);
double tanx = tan(x);
double cosxAbs = sqrt(1.0 - sinx * sinx); // по суті |cos x|

double numerator = tanx * (2.0 * sinx + 0.25 * cosxAbs);
```

Це:

- трохи оптимізує (менше операцій),
- і головне — зменшує шанс на помилку, бо формула читається простіше.

## 5. Чіткі імена змінних

У тебе і так більш-менш норм, але на майбутнє:


- `x`, `y` для координат — ок.
- `result` для дня тижня — норм, але можна `dayOfWeek`.
- Замість `white` можна `isWhiteCell` — так одразу видно, що це булевий прапор.

Чим ясніші імена — тим менше коментарів треба і тим легше змінювати код.


## 6. Коментарі: коротко й по суті

Зараз у тебе є коментар:

cpp

 Копировать код

```
// логарифм за основою 3
double log3x = log(fabs(x)) / log(3.0);
```

 LabWork number two (2)




Він нормальний, але в майбутньому:

- коментар має відповідати на «навіщо», а не просто дублювати «що робить код»;
- якщо код сам по собі вже очевидний — краще зробити вираз читабельним, ніж пояснювати його коментарем.

Наприклад:


cpp

 Копировать код

```
double logBase3 = log(fabs(x)) / log(3.0); // log_3(|x|)
```

або навіть:

cpp

 Копировать код

```
// Обчислюємо log_3(|x|) через перехід до натурального логарифма  
double logBase3 = log(fabs(x)) / log(3.0);
```



[Копировать код](#)

## 7. Структура для подібних лаб взагалі

Як шаблон для наступних робіт (де кілька завдань + математика + логіка) можна тримати в голові таку схему:

cpp

[Копировать код](#)

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

// Оголошення функцій
void taskInteger28();
void taskBoolean34();
void taskMathN11();

int main() {
    int choice;
    do {
        cout << "1 - День тижня\n"
              << "2 - Колір клітинки\n"
              << "3 - Обчислення у\n"
              << "0 - Вихід\n"
              << "Ваш вибір: ";
        cin >> choice;

        switch (choice) {
            case 1: taskInteger28(); break;
            case 2: taskBoolean34(); break;
            case 3: taskMathN11(); break;
        }
    } while (choice != 0);

    return 0;
}
```



Це не стільки про швидкість виконання, скільки про оптимізацію для людини: читати, захищати роботу, дописувати нові завдань.