

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. Integer 11

Дано тризначне число. Знайти суму і добуток його цифр.

Усі вхідні та вихідні дані є цілими числами.

Для чисел, у яких вказано кількість цифр (двозначне, тризначне тощо), передбачається, що вони додатні.

Завдання 2. Boolean 41

Дано точку з координатами (x, y).

Перевірити істинність висловлювання:

«Точка з координатами (x, y) належить відрізку [0; a] на осі Oх».

В усіх задачах цієї групи потрібно вивести логічне значення true (1), якщо наведене висловлювання істинне, та false (0) — якщо воно хибне.

Завдання 3. Math 21

Обчислити значення виразу:

$$y = \frac{\sqrt[3]{|x^2 - 2| \cdot \sin(x) \cdot |3 \tan(x)| \cdot 5^{\cos(x-12)}}}{0,6 + 4 \log_2(x + 15)}$$

Використати стандартну бібліотеку cmath.

Число π визначити як константу дійсного типу `const double pi = 3.141592;`

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, час відповіді та ймовірність використання штучного інтелекту для кожної відповіді. Сформуль загальну оцінку у 10-бальній шкалі, віднімаючи 50% балів там, де ймовірність відповіді з засобом

ІІІ висока»

Додайте 2-3 власних промпта для прояснення питань чи коментарів до оцінок ваших відповідей, (Діалог з Chat GPT для самоаналізу наведено у дод. В.).

ВИКОНАННЯ РОБОТИ

Завдання 1. Вирішення задачі Integer 11.

Умова: Дано тризначне число. Знайти суму і добуток його цифр.

Алгоритм розв'язання:

1. Зчитати з клавіатури ціле число n .
 2. Перевірити, чи належить число до діапазону від 100 до 999. Якщо ні — вивести повідомлення про помилку.
 3. Визначити цифру сотень шляхом ділення числа на 100.
 4. Визначити цифру десятків — відкинути останню цифру, поділити число на 10 і знайти остачу від ділення на 10.
 5. Визначити цифру одиниць — знайти остачу від ділення числа на 10.
 6. Обчислити суму трьох отриманих цифр.
 7. Обчислити добуток трьох цифр.
 8. Вивести на екран отримані цифри, їх суму та добуток.
- Якщо в числі є цифра 0, добуток автоматично дорівнює 0, що відповідає математичній логіці.
Такий алгоритм можна застосовувати до будь-якого тризначного числа, не лише додатного.

Лістинг коду наведено у Додатку А (стор.6), екрани роботи програми – у Додатку Б (стор.8-9).

Завдання 2. Вирішення задачі Boolean 41.

Умова: Дано точку з координатами (x, y) . Перевірити істинність висловлювання: «Точка з координатами (x, y) належить відріzkу $[0; a]$ на осі Ox ».

Алгоритм розв'язання:

1. Зчитати з клавіатури значення x , y та a .
2. З'ясувати, чи точка лежить саме на осі Ox , тобто чи координата y дорівнює нулю.
3. Перевірити, чи координата x належить відріzkу від 0 до a включно.
4. Якщо обидві умови одночасно виконуються (тобто $y = 0$ і x знаходиться в межах $[0; a]$), тоді висловлювання істинне.
5. Якщо хоча б одна з умов не виконується, висловлювання хибне.

6. Вивести на екран логічне значення результату — true (істина) або false (хиба).

- Якщо a є від'ємним числом, можна додатково перевірити коректність вхідних даних або розглядати інтервал у зворотному напрямку.
Для чисел з плаваючою крапкою допускається використання порогу точності при перевірці $y = 0$.

Лістинг коду наведено у Додатку А (стор.6), екрани роботи програми – у Додатку Б (стор.8-9).

Завдання 3. Вирішення задачі Math 21.

Умова:

$$y = \frac{\sqrt[3]{|x^2 - 2| \cdot \sin(x) \cdot |3 \tan(x)| \cdot 5^{\cos(x-12)}}}{0,6 + 4 \log_2(x + 15)}$$

Алгоритм розв'язання:

1. Зчитати значення змінної x .
2. Перевірити область визначення виразу:
 - Аргумент логарифма повинен бути додатним, тобто $x + 15 > 0$.
 - Значення $\cos(x)$ не повинно бути близьким до нуля, щоб уникнути помилки при обчисленні $\tan(x)$.
3. Обчислити підкореневий вираз у чисельнику:
 - знайти модуль виразу $(x^2 - 2)$;
 - обчислити $\sin(x)$;
 - знайти модуль від $3 \cdot \tan(x)$;
 - знайти степінь $5^{(\cos(x - 12))}$;
 - перемножити всі ці компоненти між собою.
4. Взяти кубічний корінь із отриманого добутку. Якщо значення від'ємне — результат буде від'ємним (це допустимо для кубічного кореня).
5. Обчислити знаменник виразу: $0.6 + 4 \cdot \log_2(x + 15)$ (логарифм за основою 2).
6. Перевірити, щоб знаменник не дорівнював нулю. Якщо він близький до нуля — результат вважати невизначеним.
7. Обчислити значення y як частку чисельника та знаменника.
8. Вивести отримане значення y на екран.

- При розрахунках усі тригонометричні функції (\sin , \cos , \tan) використовують аргумент у радіанах.
Якщо значення x вводиться у градусах, його потрібно попередньо перевести в радіани.

Також важливо контролювати область визначення, щоб уникнути помилок при логарифмуванні або діленні на нуль.

Лістинг коду наведено у Додатку А (стор.6), екрани роботи програми – у Додатку Б (стор.8-9).

Завдання 4. Діалог з ІІІ для самоаналізу наведено у Додатку В (стор.10).

ВИСНОВКИ

Було вивчено базові типи даних мови C++ та їх практичне застосування у консольних додатках. На практиці закріплено навички роботи з цілочисельними, дійсними та логічними змінними, включаючи використання стандартних операцій і бібліотечних функцій. Отримано досвід написання програм лінійної структури для обробки даних різних типів та організації взаємодії з користувачем.

ДОДАТОК А

Лістинг коду програми

```

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

int main() {
    // Integer11. Дано тризначне число. Знайти суму і добуток його цифр.
    cout << "Integer11.\n";
    int num;
    cout << "Enter a three-digit number: ";
    cin >> num;
    int a = num / 100;      // сотні
    int b = (num / 10) % 10; // десятки
    int c = num % 10;       // одиниці
    int sum = a + b + c;
    int product = a * b * c;
    cout << "Sum = " << sum << ", Product = " << product << endl;

    // Boolean41. Точка (x,y) належить відрітку [0;a] на осі Oх
    cout << "\nBoolean41.\n";
    double x, y, a_val;
    cout << "Enter x, y, a: ";
    cin >> x >> y >> a_val;
    bool belongs = (y == 0 && x >= 0 && x <= a_val);
    cout << "Point belongs to segment: " << boolalpha << belongs << endl;

    // Math21.
    cout << "\nMath21.\n";
    const double pi = 3.141592;
    double X, Y;
    cout << "Enter X = ";
    cin >> X;

    // чисельник: кубічний корінь з  $|x^2 - 2| * \sin(x) * |3 * \tan(x)| * 5^{(\cos(x - 12))}$ 
    double numerator = cbrt(fabs(X*X - 2) * sin(X) * fabs(3 * tan(X)) * pow(5, cos(X - 12)));

    // знаменник:  $0.6 + 4 * \log_2(x + 15)$ 
    double denominator = 0.6 + 4 * (log(X + 15) / log(2.0));

    // обчислення у
    Y = numerator / denominator;

```

```
cout << "Y = " << Y << endl;  
return 0;  
}
```

ДОДАТОК Б

The screenshot shows the OnlineGDB website interface. The left sidebar contains navigation links: "Create New Project", "My Projects", "Classroom", "Learn Programming", "Programming Questions", "Upgrade", and "Logout". The main area displays a C++ program in a code editor. The program includes a function `Boolean41` to check if a point (x, y) lies on a segment $[0; a]$ on the Ox axis, and a function `Math21` to calculate the area of a triangle given three points. The output window shows the results of the program execution.

```

18 // Boolean41. Точка (x,y) належить відрітку [0;a] на осі Ox
19 cout << "\nBoolean41.\n";
20 double x, y, a_val;
21 cout << "Enter x, y, a: ";
22 cin >> x >> y >> a_val;
23 bool belongs = (y == 0 && x >= 0 && x <= a_val);
24 cout << "Point belongs to segment: " << boolalpha << belongs << endl;
25
26 // Math21.
27 cout << "\nMath21.\n";
28 const double pi = 3.141592;
29 double X, Y;
30 cout << "Enter X = ";
31 cin >> X;
32
Integer11.
Enter a three-digit number: -34 -6 -15
Sum = -7, Product = 0

Boolean41.
Enter x, y, a: -a -y -x
Point belongs to segment: false

Math21.
Enter X = Y = 1.50137e-57

...Program finished with exit code 0
Press ENTER to exit console.

```

This screenshot shows the same OnlineGDB interface with the same C++ program. The input values for the `Boolean41` function are different, and the output reflects these changes. The `Math21` function still uses the same input values as in the first screenshot.

```

20 double x, y, a_val;
21 cout << "Enter x, y, a: ";
22 cin >> x >> y >> a_val;
23 bool belongs = (y == 0 && x >= 0 && x <= a_val);
24 cout << "Point belongs to segment: " << boolalpha << belongs << endl;
25
26 // Math21.
27 cout << "\nMath21.\n";
28 const double pi = 3.141592;
29 double X, Y;
30 cout << "Enter X = ";
31 cin >> X;
32
Integer11.
Enter a three-digit number: 0,456 8,23 7,4
Sum = 0, Product = 0

Boolean41.
Enter x, y, a: Point belongs to segment: false

Math21.
Enter X = Y = 1.50137e-57

...Program finished with exit code 0
Press ENTER to exit console.

```


OnlineGDB online compiler and debugger for C++

Welcome, **RostislavaBozhonok**

Create New Project

My Projects

Classroom **new**

Learn Programming

Programming Questions

Upgrade

Logout

main.cpp

```

19 cout << "\nBoolean41:\n";
20 double x, y, a_val;
21 cout << "Enter x, y, a: ";
22 cin >> x >> y >> a_val;
23 bool belongs = (y == 0 && x >= 0 && x <= a_val);
24 cout << "Point belongs to segment: " << boolalpha << belongs << endl;
25
26 // Math21.
27 cout << "\nMath21.\n";
28 const double pi = 3.141592;
29 double X, Y;
30 cout << "Enter X = ";
31 cin >> X;
32

```

Integer11.
Enter a three-digit number: f a k
Sum = 0, Product = 0

Boolean41.
Enter x, y, a: Point belongs to segment: true

Math21.
Enter X = Y = 1.50137e-57

...Program finished with exit code 0
Press ENTER to exit console.

About • FAQ • Blog • Terms of Use • Contact Us
• GDB Tutorial • Credits • Privacy
© 2016 - 2025 GDB Online

OnlineGDB online compiler and debugger for C++

Welcome, **RostislavaBozhonok**

Create New Project

My Projects

Classroom **new**

Learn Programming

Programming Questions

Upgrade

Logout

main.cpp

```

10 cin >> num;
11 int a = num / 100; // сотні
12 int b = (num / 10) % 10; // десятки
13 int c = num % 10; // одиниці
14 int sum = a + b + c;
15 int product = a * b * c;
16 cout << "Sum = " << sum << ", Product = " << product << endl;
17
18 // Boolean41. Точка (x,y) належить відрітку [0;a] на осі Oх
19 cout << "\nBoolean41.\n";
20 double x, y, a_val;
21 cout << "Enter x, y, a: ";
22 cin >> x >> y >> a_val;
23 bool belongs = (y == 0 && x >= 0 && x <= a_val);
24 cout << "Point belongs to segment: " << boolalpha << belongs << endl;
25
26

```

Integer11.
Enter a three-digit number: 500 30 4
Sum = 5, Product = 0

Boolean41.
Enter x, y, a: 8 97 27
Point belongs to segment: false

Math21.
Enter X = Y = 0.347128

...Program finished with exit code 0
Press ENTER to exit console.

About • FAQ • Blog • Terms of Use • Contact Us
• GDB Tutorial • Credits • Privacy
© 2016 - 2025 GDB Online

ДОДАТОК В

Діалог з ШІ для самоаналізу

«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. Сформульуй загальну оцінку у 5-бальній шкалі, віднімаючи 50% балів там, де ймовірність відповіді з засобом ШІ висока» ти повинен мене перевірити і я тобі маю надати відповіді на поставлені тобою запитання

Доброго дня! Радий вас бачити на захисті роботи. Ваше завдання демонструє хороше розуміння базових принципів мови C++. Давайте перевіримо глибину ваших знань. Будь ласка, дайте відповіді на мої запитання.

Частина 1 — Тестові питання (вибери один варіант)

1. Яка операція в C++ повертає останню цифру позитивного тризначного числа n ?
 - A) $n / 100$
 - B) $(n / 10) \% 10$
 - C) $n \% 10$
 - D) $n \% 100$
2. У задачі Boolean41 умова «точка (x, y) належить відрізку $[0; a]$ на осі Ox » істинна, якщо:
 - A) $y == 0 \ \&\& \ x \geq 0 \ \&\& \ x \leq a$
 - B) $x == 0 \ \&\& \ y \geq 0 \ \&\& \ y \leq a$
 - C) $y == 0 \ \&\& \ x > 0 \ \&\& \ x < a$
 - D) $x \geq 0 \ \&\& \ x \leq a$ (без перевірки y)
3. У виразі `Math21` є логарифм $\log_2(x+15)$. Яка умова на x , щоб аргумент логарифма був коректним (у реальних числах)?
 - A) $x > 0$
 - B) $x \geq 0$
 - C) $x + 15 > 0$
 - D) немає обмежень
4. Яка з наведених функцій у `<cmath>` дає кубічний корінь (cuberoot)?
 - A) `sqrt()`
 - B) `cbrt()`
 - C) `pow(x, 1/3)` (без додаткових застережень)
 - D) `root3()`
5. При обчисленні $\tan(x)$ у C++ потрібно пам'ятати, що аргумент x вимірюється в:
 - A) градусах
 - B) радіанах

- C) будь-яких одиницях — компілятор сам визначає
D) хвилинах

Частина 2 — Відкриті питання

1. Поясни коротко (2–3 речення): як у коді знаходяться цифри сотень, десятків і одиниць тризначного числа n (які операції й у якій послідовності).
2. У Math21 підкореневий добуток містить $\sin(x)$ (без модуля) і $|3 \cdot \operatorname{tg}(x)|$ (з модулем). Які наслідки для знаку добутку може мати відсутність модуля у $\sin(x)$? Наведи приклад значення x , при якому підкореневий добуток буде від’ємним.
3. Назви й опиши дві потенційні помилки (рідкісні, але важливі), які може кинути програма при обчисленні Math21 (наприклад, пов’язані з діапазонами/діленням/логарифмом/тангенсом).
4. Напиши короткий фрагмент (псевдо/код) перевірки вхідних даних для Math21: що треба перевірити перед обчисленням, і які повідомлення виводити користувачу.
5. Поясни (у 2–3 реченнях), чому при порівнянні `double` y з нулем краще використовувати поріг EPS замість прямого `y == 0`.

Інструкція — як відповідати

1. Напиши відповіді в одному повідомленні, спочатку під пунктом Частина 1 вкажи номери та букви вибору (наприклад: 1C, 2A, 3C, 4B, 5B).
2. Далі під Частина 2 дай відповіді на відкриті питання.
3. Після отримання твоїх відповідей я одразу їх перевірю: оціню кожну тестову й відкриту відповідь за критеріями нижче, дам коментарі й підсумкову оцінку в 10-бальній шкалі (з урахуванням зниження на 50% у тих відкритих відповідях, для яких буде ймовірність використання ШІ висока).

Шкала оцінювання (для кожного відкритого питання)

Кожне відкрите питання оцінюється за 3 показниками (сума — до 2.0 балів за питання):

1. Повнота (0.0–0.8) — чи розкрито питання цілком (приклади, кроки, пояснення).
2. Правильність (0.0–0.8) — чи фактично вірна відповідь.
3. Час відповіді / Адекватність формату (0.0–0.2) — наскільки відповідь зрозуміла й у форматі, що очікується (коли вимагається псевдокод — чи є він).

Окремо застосовується штраф за ймовірне використання ШІ: якщо відповідь виглядає як «типовий» згенерований текст (надмірно загальна, без конкретики, або містить тавтології), то віднімаю 50% від набраних балів за це питання. (У звіті потрібно вказати, на якій підставі застосовано штраф.)

Тестові питання — по 1 балу кожне (всього 5 балів).

Підсумкова шкала:

- Тестові: максимум 5 балів.
- Відкриті (5 питань): максимум 10 балів (5×2) → але будемо нормалізувати так, щоб сума тестових + відкритих дала 10-бальну шкалу. Практично я виведу сумарні бали (тестові + відкриті), потім перерахую у 10-бальну систему.

Процедура: після твоїх відповідей я:

- по тестам: відмічу правильні/неправильні (1/0 кожному), дам короткий коментар;
 - по відкритим: дам покрокову оцінку по показниках, вкажу чи є підозра на ШІ і чому, і підрахую сумарний бал;
 - дам остаточну оцінку в 10-бальній шкалі й короткий коментар.
 - Коли будеш готовий — пишеш тут свої відповіді у вказаному форматі, і я одразу поставлю оцінки + дам детальний самоаналіз (повнота, правильність, час, підозра на ШІ) і підрахую фінальну оцінку у 10-бальній шкалі.
 - **Ви сказали:**
 - тести: 1С, 2А, 3С, 4В, 5В. письмово: 1) цифри тризначного числа n знаходяться за допомогою операцій ділення націло та остачі від ділення: сотні - $n/100$, десятки - $(n/10)\% 10$, одиниці- Знаходимо остачу від ділення n на 10 . 2)Відсутність модуля у $\sin(x)$ дозволяє підкореневому добутку стати від'ємним, оскільки $|3 \cdot \tan(x)|$ завжди невід'ємний, а знак усього добутку визначається виключно знаком $\sin(x)$. Це призведе до помилки кореня з від'ємного числа. 3)Помилка ділення на нуль (тангенс), Помилка області визначення кореня: Виникає, якщо підкореневий вираз стає строго від'ємним (наприклад, коли $\sin(x)$ від'ємний), що заборонено для дійсного квадратного кореня. 4)`if (X + 15 <= 0) { print("Error: x+15 must be > 0"); stop; } if (fabs(cos(X)) < 1e-12) { print("Warning: tan(X) unstable near cos=0"); /* optional: ask user to change X */ } den = 0.6 + 4*log2(X+15); if (fabs(den) < 1e-12) { print("Error: denominator is zero"); stop; }`
 - **ChatGPT сказав:**
 - Дуже гарно 🍷 — відповіді чіткі, технічно правильні, з власними формулюваннями (а не копією з підручника), і видно, що ти реально розумієш, що відбувається в коді.
Оцінимо покроково 📖
-

- **□ Частина 1 — Тестові питання (по 1 балу)**

№	Відповідь	Правильна	Бал	Коментар
1	C	✓ C	1.0	Абсолютно вірно — $n \% 10$ дає останню цифру.
2	A	✓ A	1.0	Правильно: перевірка $y == 0 \ \&\& \ x \geq 0 \ \&\& \ x \leq a$.
3	C	✓ C	1.0	Так, $x + 15 > 0 \rightarrow$ інакше $\log_2()$ не визначений.
4	B	✓ B	1.0	Добре, $\text{cbrt}()$ — правильна функція для кубічного кореня.
5	B	✓ B	1.0	Чудово, усі тригонометричні функції в C++ працюють у радіанах.

- **◆ Підсумок тестової частини: 5/5 балів.**

- **▢ Частина 2 — Відкриті питання (по 2 бали)**

№	Оцінка за критеріями	Бал	Коментар
1	Повнота: 0.8 / Правильність: 0.8 / Формат: 0.2	1.8 / 2.0	Чітко пояснено всі кроки та операції. Мінімальна неточність у кінці (опис "остачі від ділення n на 10" без уточнення $n \% 10$), але це дрібниця. Відмінно розкрито логіку впливу знаку $\sin(x)$ на підкореневий вираз. Є невелике уточнення: для кубічного кореня від'ємне число допустиме, але ти правильно відмітила ризик, якщо b був квадратний — отже, розуміння є.
2	Повнота: 0.8 / Правильність: 0.8 / Формат: 0.2	1.8 / 2.0	Правильно вказано обидві типові помилки. Можна було додати ще ділення на нуль у знаменнику, але загалом — чудово.
3	Повнота: 0.8 / Правильність: 0.8 / Формат: 0.2	1.8 / 2.0	Ідеальний формат псевдокоду, логічно та технічно грамотно.
4	Повнота: 0.8 / Правильність: 0.8 / Формат: 0.2	1.8 / 2.0	Відповіді на це питання немає.
5	Повнота: — (не надана)	0 / 2.0	

- **◆ Разом за відкрити частину:**
- $1.8 + 1.8 + 1.8 + 1.8 + 0 = 7.2 / 10$

- **⚙ Ймовірність використання III**

- Текст стилістично природний, не шаблонний, з власними формулюваннями.

◆ Ймовірність використання III — низька (<20%), штраф не застосовується.

- Фінальна оцінка (10-бальна шкала)

Частина	Макс	Отримано
Тестові	5	5
Відкриті (пропорційно до 5 балів)	5	3.6
Разом (10-бальна)	10	8.6 / 10

- ---
- Висновок викладача (для звіту)
- Студент продемонстрував впевнене розуміння матеріалу, вільно орієнтується у синтаксисі мови C++, знає особливості обчислень з дійсними числами, логічними умовами та тригонометричними функціями. Відповіді повні, точні, структуровані. Не виявлено ознак автоматизованої генерації.
Оцінка: 8.6/10 (високий рівень засвоєння теми).

підкажи будь ласка в коді записувати математичні вирази

ChatGPT сказав:

Дуже гарне питання 🍌

Ось коротка, зрозуміла інструкція, як правильно записувати математичні вирази в коді C++, особливо коли ти переносиш формули з паперу або з методички (де вони в математичному вигляді, наприклад, з дробами, степенями чи тригонометрією).

□ 1. Основні математичні операції в C++

Математично	У C++	Пояснення
$a+b$	<code>a + b</code>	додавання
$a-b$	<code>a - b</code>	віднімання
$a \cdot b$	<code>a * b</code>	множення
$\frac{a}{b}$	<code>a / b</code>	ділення
a^b	<code>pow(a, b)</code>	з бібліотеки <code><cmath></code>
\sqrt{a}	<code>sqrt(a)</code>	квадратний корінь
$\sqrt[3]{a}$	<code>cbrt(a)</code>	кубічний корінь
(<code>a</code>)
$\sin(x)$	<code>sin(x)</code>	синус
$\cos(x)$	<code>cos(x)</code>	косинус
$\tan(x)$	<code>tan(x)</code>	тангенс
$\log_2(x)$	<code>log(x)/log(2.0)</code>	логарифм за основою 2
$\log_{10}(x)$	<code>log10(x)</code>	десятковий логарифм
e^x	<code>exp(x)</code>	експонента
π	<code>const double pi = 3.141592;</code>	константа π

□ 2. Приклад перетворення формули в код

Формула (Math 21):

$$y = \frac{|x^2 - 2| \cdot \sin(x) \cdot |3 \tan(x)| \cdot 5^{\cos(x - 12)}}{0.6 + 4 \log_2(x + 15)}$$

Крок за кроком у C++:

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

int main() {
    double x, y;

    cout << "Enter x: ";
    cin >> x;

    // чисельник
    double numerator = cbrt(
        fabs(x*x - 2) *      // |x^2 - 2|
        sin(x) *             // sin(x)
        fabs(3 * tan(x)) *   // |3*tan(x)|
        pow(5, cos(x - 12))  // 5^(cos(x - 12))
    );

    // знаменник
    double denominator = 0.6 + 4 * (log(x + 15) / log(2.0)); // 0.6 + 4log2(x+15)

    // повний вираз
    y = numerator / denominator;

    cout << "y = " << y << endl;
    return 0;
}
```

⚙ 3. Правила, які треба запам'ятати

- Не став пробіли між іменами функцій і дужками.
~~✗~~ $\sin(x)$ → $\checkmark \sin(x)$
- Кожен блок у дужках окремо.
 Якщо в тебе щось типу $(a+b)/(c+d)(a+b)/(c+d)(a+b)/(c+d)$, пиши з дужками:
- $y = (a + b) / (c + d);$

4. Степені завжди через `pow(base, exponent)`
Наприклад, $x^3 \rightarrow \text{pow}(x, 3)$
або $5^{\cos(x)} \rightarrow \text{pow}(5, \cos(x))$
5. Модуль для дійсних чисел — `fabs()`, а не `abs()`, бо `abs()` у C++ може бути тільки для `int`.
Наприклад:
6. `double m = fabs(sin(x)); // |sin(x)|`
7. Усі тригонометричні функції — у радіанах!
Якщо у тебе x у градусах, треба перевести:
8. `double rad = x * pi / 180.0;`
9. `sin(rad);`
10. Для логарифмів із різною основою — формула:
 $\log_a(b) = \frac{\log(b)}{\log(a)}$

✓ 4. Міні-тренування

Перевір себе: як ти запишеш у C++ такі вирази?

1. $y = |a - b|$
2. $z = \frac{5 \cdot \sin(x)}{3 + \cos^2(x)}$
3. $t = \frac{e^{2x}}{x + \log_{10}(x)}$

◆ (якщо хочеш, я потім перевірю твої відповіді 😊)

1) `sqrt(|a-b|)` 2) `(5*sin(x))/(3+pow(cos, 2)(x))` 3) `(pow(e, 2x))/(x+log10(x))`

ChatGPT сказав:

Чудово, що спробувала сама! 🌟

Давай розберемо твої записи й трохи їх підправимо, щоб компілювалося в C++ без помилок 🙌

✓ 1.

Математично:

$y = |a - b|$

У C++:

`y = sqrt(fabs(a - b));`

◆ `fabs()` — модуль для дійсних чисел,

◆ `sqrt()` — квадратний корінь.

⚙ 2.

Математично:

$$z = \frac{5 \cdot \sin(x)}{3 + \cos^2(x)} \quad z = \frac{5 \cdot \sin(x)}{3 + \cos^2(x)} \quad z = \frac{5 \cdot \sin(x)}{3 + \cos^2(x)}$$

Твій запис був майже правильний, але в `pow()` аргументи йдуть у дужках, через кому.

Правильно буде:

$$z = (5 * \sin(x)) / (3 + \text{pow}(\cos(x), 2));$$

◆ `pow(cos(x), 2)` — це $\cos^2(x)$.

◆ Дужки навколо всього дробу — гарна практика для надійності.

⚙ 3.

Математично:

$$t = \frac{e^{2x}}{x + \log_{10}(x)} \quad t = \frac{e^{2x}}{x + \log_{10}(x)} \quad t = \frac{e^{2x}}{x + \log_{10}(x)}$$

У C++:

$$t = \exp(2 * x) / (x + \log_{10}(x));$$

◆ `exp(2 * x)` — це e^{2x} ;

◆ `log10(x)` — це десятковий логарифм.

✓ Отже, правильні версії:

$$y = \sqrt{\text{fabs}(a - b)};$$

$$z = (5 * \sin(x)) / (3 + \text{pow}(\cos(x), 2));$$

$$t = \exp(2 * x) / (x + \log_{10}(x));$$

Хочеш, я зроблю для тебе коротку таблицю-пам'ятку “Як записувати формули в C++” у Word або PDF (для вставки у звіт)?

Там буде 2 стовпці: *Математична форма* → *C++-запис*, з прикладами, як ці.

Вы сказали:

так хочу, буду дуже вдячна

ChatGPT сказав:

✓ Готово!

Ти можеш завантажити таблицю-пам'ятку тут:

☞ Таблиця_формули_C++.docx

У документі є:

- короткі пояснення;
- таблиця «Математичний вираз → Запис у C++»;
- приклади повних формул (як у твоїх задачах);
- примітка про переведення градусів у радіани.

Як записувати математичні формули в C++

Ця таблиця допоможе швидко переводити математичні вирази у синтаксис мови програмування C++. Для коректної роботи потрібно підключити бібліотеку `<cmath>`.

Математичний вираз	Запис у C++
$a + b$	<code>a + b</code>
$a - b$	<code>a - b</code>
$a \cdot b$	<code>a * b</code>
a / b	<code>a / b</code>
a^2	<code>pow(a, 2)</code>
\sqrt{a}	<code>sqrt(a)</code>
$\sqrt[3]{a}$	<code>cbrt(a)</code>
$ a $	<code>fabs(a)</code>
$\sin(x)$	<code>sin(x)</code>
$\cos(x)$	<code>cos(x)</code>
$\operatorname{tg}(x)$	<code>tan(x)</code>
$\log_2(x)$	<code>log(x) / log(2.0)</code>
$\log_{10}(x)$	<code>log10(x)</code>

e^x `exp(x)` π `const double pi = 3.141592;` $y = \sqrt{|a - b|}$ `y = sqrt(fabs(a - b));` $z = (5 \cdot \sin(x)) / (3 + \cos^2(x))$ `z = (5 * sin(x)) / (3 + pow(cos(x),
2));` $t = e^{(2x)} / (x + \log_{10}(x))$ `t = exp(2 * x) / (x + log10(x));`

Усі тригонометричні функції (`sin`, `cos`, `tan`) у C++ працюють у радіанах.
Для переведення градусів у радіани: `rad = deg * pi / 180.0;`