

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний аерокосмічний університет

«Харківський авіаційний інститут»

Факультет систем управління літальних апаратів

Кафедра систем управління літальних апаратів

Лабораторна робота № 4

з дисципліни «Алгоритмізація та програмування»

на тему «Структурування програм з використанням функцій»

XAI.301. G3. 319a. 1 ЛР

Виконав студент гр. 319a

Вікторія БАБІНА

(підпис, дата)

(П.І.Б.)

Перевірив

асис. Євгеній ПЯВКА

(підпис, дата)

(П.І.Б.)

2025

МЕТА РОБОТИ

Вивчити теоретичний матеріал із синтаксису оголошення, визначення і виклику функцій в C ++ і реалізувати консольний додаток з використанням функцій з параметрами і поверненням результата на мові програмування C++ в середовищі Qt Creator.

ПОСТАНОВКА ЗАДАЧІ

Завдання 1. Вирішити дві задачі з реалізації функцій. Дляожної задачі описати функцію (декларація, визначення і виклик) відповідно до варіанту. У тілі і при виклику функцій не використовувати цикли.

Proc3. Описати процедуру Mean (X, Y, AMean, GMean), яка обчислює середнє арифметичне $AMean = (X + Y) / 2$ і середнє геометричне $GMean = (X \cdot Y)^{1/2}$ двох додатних чисел X і Y (X і Y - вхідні, AMean і GMean - вихідні параметри дійсного типу). За допомогою цієї процедури знайти середнє арифметичне і середнє геометричне для пар (A, B), (A, C), (A, D), якщо дані A, B, C, D.

Proc51. Описати функцію TriangleType(a, b, c) цілого типу, яка повертає: 1- якщо трикутник гострокутний, 0 - якщо прямокутний, -1 - якщо тупокутний. За допомогою цієї функції визначити типи трьох трикутників з даними сторонами.

Завдання 2. Для вирішення завдання з ціличисельними змінними з Лабораторної роботи №2 відповідно до варіанту визначити три функції:

- 1) функцію введення вхідних значень з консолі з перевіркою їх на коректність;
- 2) функцію підрахунку результату;
- 3) функцію виведення результату в консоль.

При виклику функцій в разі введення некоректних вхідних даних вивести відповідне повідомлення і розрахунки не виконувати.

Integer21. З початку доби минуло N секунд (N - ціле). Знайти кількість секунд, що пройшли з початку останньої хвилини.

Завдання 3. Для вибору користувачем одного з трьох зазначених вище завдань організувати меню з використанням інструкції вибору. Кожне завдання має бути реалізовано у вигляді окремої процедури (функції без параметрів), що містить необхідні оголошення змінних і виклики інших функцій.

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

- 1) «Ти - викладач, що приймає захист моєї роботи. Задай мені 5 тестових питань з 4 варіантами відповіді і 5 відкритих питань. Це мають бути завдання <середнього> рівня складності на розвиток критичного та інженерного мислення. Питання мають відноситись до коду, що є у файлі звіту, і до теоретичних відомостей, що є у файлі лекції»
 - 2) «Проаналізуї повноту, правильність відповіді та ймовірність використання штучного інтелекту для кожної відповіді. Оціни кожне питання у 5-балльній шкалі, віднімаючи 60% балів там, де ймовірність відповіді з засобом ШІ висока. Обчисли загальну середню оцінку»
 - 3) «Проаналізуї код у звіті, і додай опис і приклади коду з питань, які є в теоретичних відомостях, але не відпрацьовано у коді при вирішенні завдань»
- Проаналізуйте задані питання, коментарі і оцінки, надані ШІ. Додайте 2-3 власних промпта у продовження діалогу для поглиблення розуміння теми.

ВИКОНАННЯ РОБОТИ

Завдання 1.

Вирішення задачі Proc3

Вхідні дані (ім'я, опис, тип, обмеження):

A-перше додатне число, double, A > 0

B-друге додатне число, double, B > 0

C-третє додатне число, double, C > 0

D-четверте додатне число, double, D > 0

X-перше число для процедури Mean, double, X > 0

Y-друге число для процедури Mean, double, Y > 0

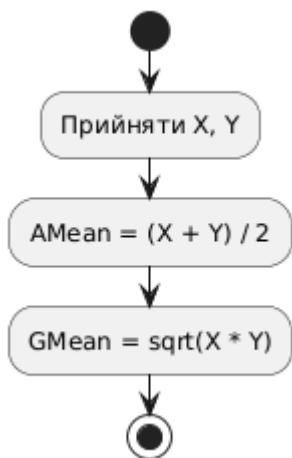
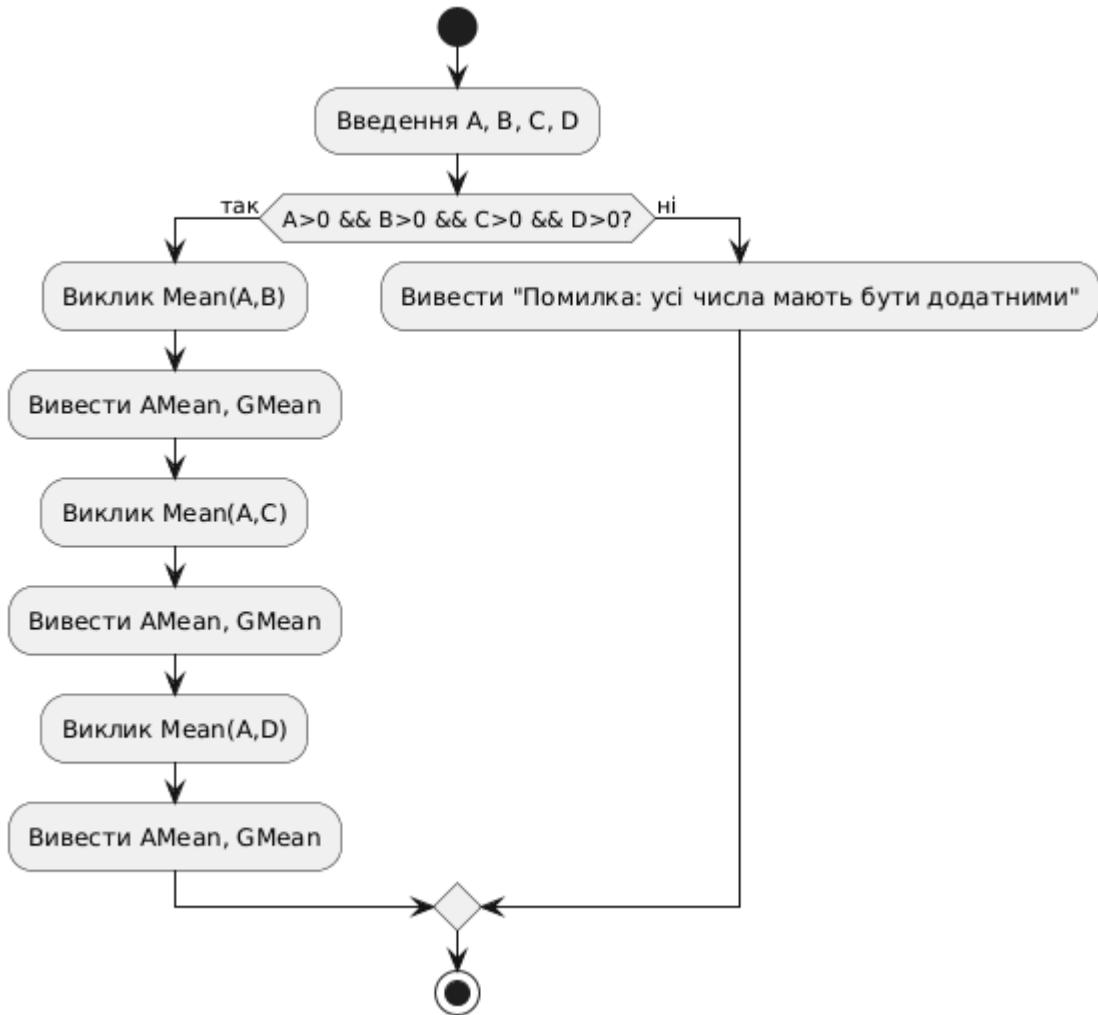
Вихідні дані (ім'я, опис, тип):

AMean- середнє арифметичне чисел X і Y, double

GMean-середнє геометричне чисел X і Y, double

Текстові повідомлення-результати обчислення для пар (A,B), (A,C), (A,D), string

Алгоритм вирішення показано нижче



Вирішення задачі Proc51

Вхідні дані (ім'я, опис, тип, обмеження):

a- перша сторона трикутника, double, $a > 0$

b-друга сторона трикутника, double, $b > 0$

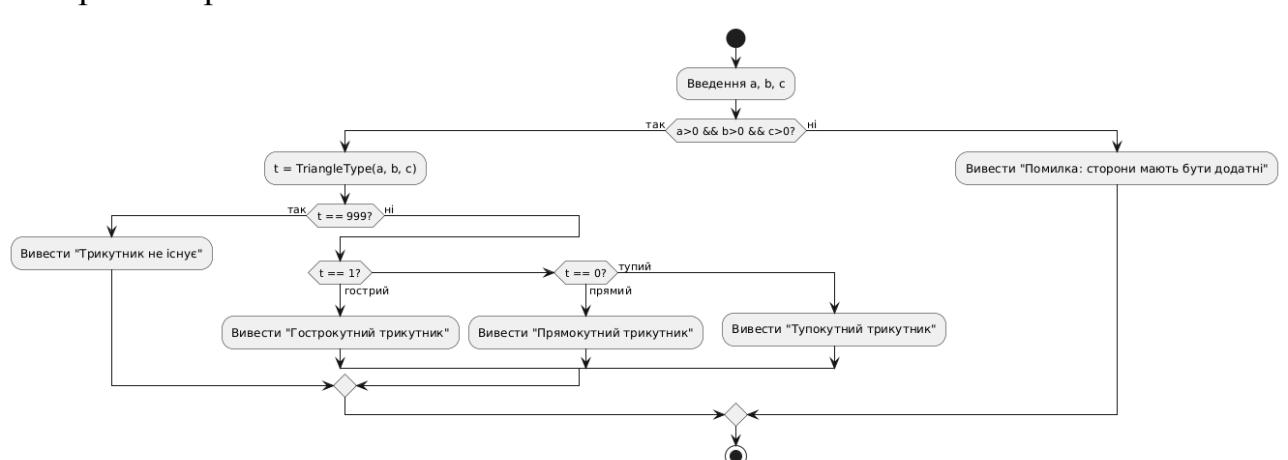
c-третя сторона трикутника, double, $c > 0$

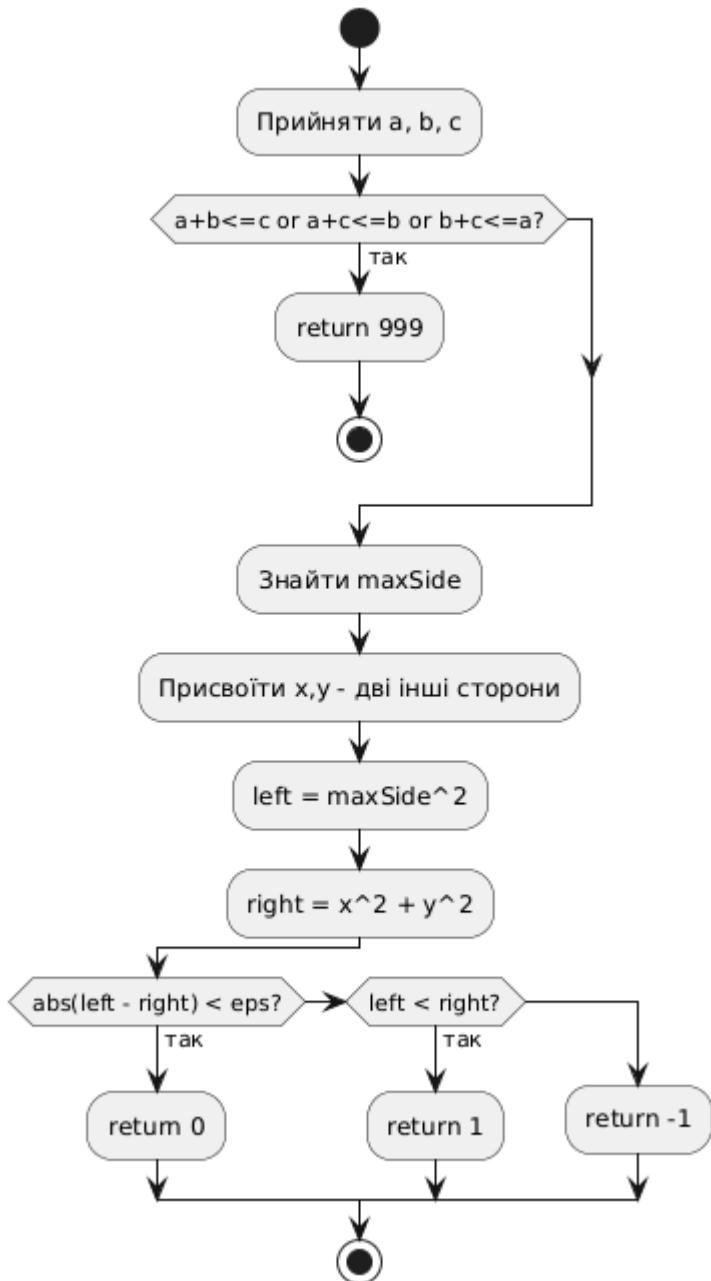
Вихідні дані (ім'я, опис, тип):

t -тип трикутника: 1 – гострокутний, 0 – прямокутний, -1 – тупокутний, 999 – не існує, int

Текстові повідомлення-інформація про тип трикутника, string

Алгоритм вирішення показано нижче





Лістинг коду вирішення задачі Proc3 та Proc51 наведено в дод. А (стор. 10). Екран роботи програми показаний на рис. Б.1 та Б.2.

Завдання 2.

Вирішення задачі Integer21

Вхідні дані (ім'я, опис, тип, обмеження):

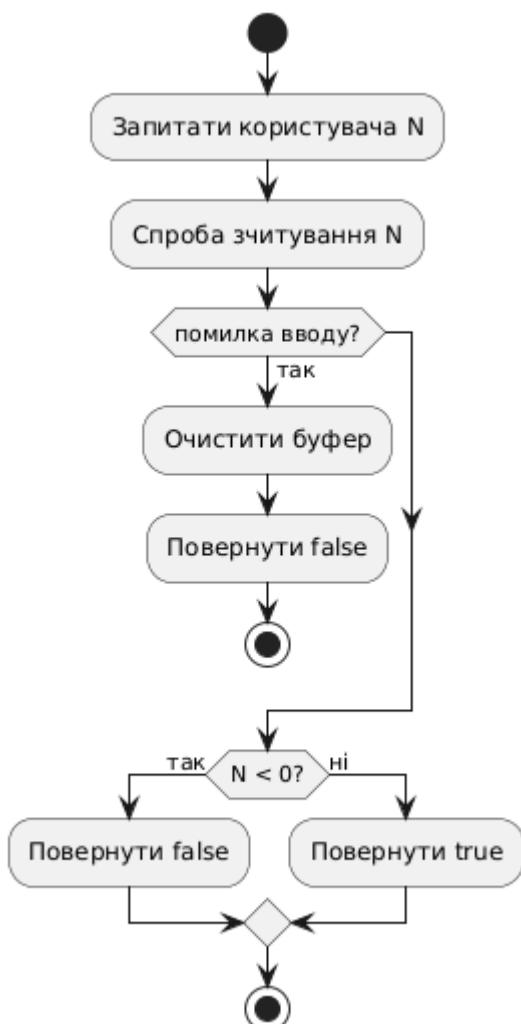
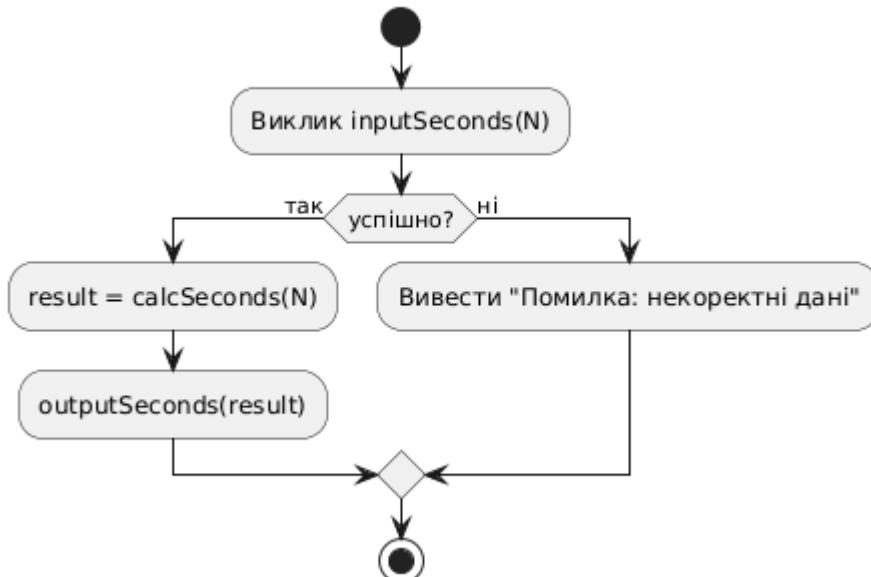
N - кількість секунд, що минули з початку доби, int, $N \geq 0$

Вихідні дані (ім'я, опис, тип):

seconds- кількість секунд, що пройшли з початку останньої хвилини, int

Текстові повідомлення - повідомлення про некоректне введення або результат роботи програми, string

Алгоритм вирішення показано нижче



Лістинг коду вирішення задачі Integer21 наведено в дод. А (стор. 10).

Екран роботи програми показаний на рис. Б.1.

Завдання 4.

1. C 2. D 3. B 4. B 5. B

1) Параметри AMean і GMean — це вихідні значення, які потрібно повернути з процедури. Через посилання ми передаємо змінні напряму, і процедура може змінити їхній вміст без повернення через return. Переваги: можна повернути одразу два результати; не потрібно створювати додаткові структури або глобальні змінні; працює швидше, бо не створюються копії змінних.

2) Тому що користувач може ввести не число, наприклад текст або символи. У цьому випадку сін переходить в стан помилки, і подальше читання стане неможливим.

`cin.fail()` → виявляє помилку вводу

`cin.clear()` → скидає стан помилки

`cin.ignore()` → очищає залишки неправильного вводу

Без цього програма просто застрягла б і далі не працювала.

3) За теоремою косинусів:

для прямокутного трикутника виконується $c^2 = a^2 + b^2$

де c — найбільша сторона.

У коді знаходиться найбільша сторона maxSide, а потім порівнюється. Якщо рівність виконується (з точністю до $1e-6$) → трикутник прямокутний.

4) Бо будь-яке неправильне значення може зламати обчислення або дати безглуздий результат.

5) Найпростіший спосіб — використовувати масив вказівників на функції або `std::map<int, function>`. Тоді при додаванні нового пункту меню треба лише: додати функцію в масив, додати назву у список. Меню саме викликатиме відповідну функцію без купи case.

ВИСНОВКИ

Були вивчені та закріплені на практиці принципи створення і використання функцій та процедур у мові C++. Відпрацьовано у коді передачу параметрів, перевірку коректності вводу та обробку різних типів задач. Отримано навички побудови меню програми та розподілу логіки на окремі функції, хоча певні труднощі виникали під час організації перевірки вводу і реалізації алгоритму визначення типу трикутника.

ДОДАТОК А

Лістинг коду програми

```

#include <iostream>
#include <cmath> // для математичних функцій
using namespace std;

// Прототипи
void task_Proc3();      // Завдання 1 – Proc3: Mean(X, Y, AMean, GMean)
void task_Integer21();   // Завдання 2 – Integer21
void task_Proc51();      // Завдання 3 – Proc51: TriangleType(a, b, c)
void showMenu();         // Меню вибору користувача

// Підфункції для Integer21
bool inputSeconds(int &N);    // введення з перевіркою
int calcSeconds(int N);       // обчислення
void outputSeconds(int seconds); // виведення результату

// Підфункції для Proc3
void Mean(double X, double Y, double &AMean, double &GMean);

// Підфункції для Proc51
int TriangleType(double a, double b, double c);

int main() {
    int choice;
    do {
        showMenu();
        cout << "Оберіть завдання (1-3) або 0 для виходу: ";
        cin >> choice;
        cout << endl;

        switch (choice) {
            case 1: task_Proc3(); break;
            case 2: task_Integer21(); break;
            case 3: task_Proc51(); break;
            case 0: cout << "Вихід із програми.\n"; break;
            default: cout << "Невірний вибір. Спробуйте ще раз.\n";
        }
        cout << endl;
    } while (choice != 0);

    return 0;
}

// Визначення функцій

// Завдання 1: Proc3
void Mean(double X, double Y, double &AMean, double &GMean) {
    AMean = (X + Y) / 2;
}

```

```

GMean = sqrt(X * Y);
}

void task_Proc3() {
    cout << " Завдання 1: Proc3 (Mean) \n";
    double A, B, C, D;
    cout << "Введіть 4 додатних числа A, B, C, D: ";
    cin >> A >> B >> C >> D;

    if (A <= 0 || B <= 0 || C <= 0 || D <= 0) {
        cout << "Помилка: усі числа мають бути додатними.\n";
        return;
    }

    double AMean, GMean;
    Mean(A, B, AMean, GMean);
    cout << "(A,B): AMean=" << AMean << ", GMean=" << GMean << endl;

    Mean(A, C, AMean, GMean);
    cout << "(A,C): AMean=" << AMean << ", GMean=" << GMean << endl;

    Mean(A, D, AMean, GMean);
    cout << "(A,D): AMean=" << AMean << ", GMean=" << GMean << endl;
}

// Завдання 2: Integer21
// 1. Введення даних
bool inputSeconds(int &N) {
    cout << "Введіть кількість секунд з початку доби: ";
    cin >> N;
    if (cin.fail() || N < 0) {
        cin.clear();
        cin.ignore(1000, '\n');
        return false;
    }
    return true;
}

// 2. Обчислення результату
int calcSeconds(int N) {
    return N % 60; // залишок секунд з останньої хвилини
}

// 3. Виведення результату
void outputSeconds(int seconds) {
    cout << "Кількість секунд з початку останньої хвилини: " << seconds << endl;
}

// 4. Головна процедура для Integer21
void task_Integer21() {

```

```

cout << " Завдання 2: Integer21 \n";
int N;
if (!inputSeconds(N)) {
    cout << "Помилка: введено некоректні дані!\n";
    return;
}

int result = calcSeconds(N);
outputSeconds(result);
}

// Завдання 3: Proc51
int TriangleType(double a, double b, double c) {
    if (a + b <= c || a + c <= b || b + c <= a)
        return 999; // трикутник не існує

    double maxSide = max(a, max(b, c));
    double sumSq = a * a + b * b + c * c - maxSide * maxSide;

    // Теорема косинусів
    if (fabs(a * a + b * b + c * c - 2 * maxSide * maxSide) < 1e-6)
        return 0; // прямокутний
    else if (a * a + b * b + c * c > 2 * maxSide * maxSide)
        return 1; // гострокутний
    else
        return -1; // тупокутний
}

void task_Proc51() {
    cout << " Завдання 3: Proc51 (TriangleType) \n";
    double a, b, c;
    cout << "Введіть сторони трикутника a, b, c: ";
    cin >> a >> b >> c;

    if (a <= 0 || b <= 0 || c <= 0) {
        cout << "Помилка: сторони мають бути додатні.\n";
        return;
    }

    int t = TriangleType(a, b, c);
    if (t == 999)
        cout << "Трикутник не існує.\n";
    else if (t == 1)
        cout << "Трикутник гострокутний.\n";
    else if (t == 0)
        cout << "Трикутник прямокутний.\n";
    else
        cout << "Трикутник тупокутний.\n";
}

```

```
// ===== Меню =====
void showMenu() {
    cout << "-----\n";
    cout << "      МЕНЮ ЛР №4      \n";
    cout << "1 - Proc3\n";
    cout << "2 - Integer21\n";
    cout << "3 - Proc51\n";
    cout << "0 - Вихід\n";
    cout << "-----\n";
}
```

ДОДАТОК Б

Скрін-шоти вікна виконання програми

```

МЕНЮ ЛР №4
1 - Proc3
2 - Integer21
3 - Proc51
0 - Вихід
-----
Оберіть завдання (1-3) або 0 для виходу: 2

Завдання 2: Integer21
Введіть кількість секунд з початку доби: 7
Кількість секунд з початку останньої хвилини: 7

-----
МЕНЮ ЛР №4
1 - Proc3
2 - Integer21
3 - Proc51
0 - Вихід
-----
Оберіть завдання (1-3) або 0 для виходу: 1

Завдання 1: Proc3 (Mean)
Введіть 4 додатніх числа A, B, C, D: 3 5 7 9
(A,B) : AMean=4, GMean=3.87298
(A,C) : AMean=5, GMean=4.58258
(A,D) : AMean=6, GMean=5.19615

-----
МЕНЮ ЛР №4
1 - Proc3

```

Рисунок Б.1 – Екран виконання програми для вирішення завдання Integer21 та Proc3

```

МЕНЮ ЛР №4
0 - Вихід
-----
Оберіть завдання (1-3) або 0 для виходу: 1

Завдання 1: Proc3 (Mean)
Введіть 4 додатніх числа A, B, C, D: 3 5 7 9
(A,B) : AMean=4, GMean=3.87298
(A,C) : AMean=5, GMean=4.58258
(A,D) : AMean=6, GMean=5.19615

-----
МЕНЮ ЛР №4
1 - Proc3
2 - Integer21
3 - Proc51
0 - Вихід
-----
Оберіть завдання (1-3) або 0 для виходу: 3

Завдання 3: Proc51 (TriangleType)
Введіть сторони трикутника a, b, c: 3 5 7
Трикутник тупокутний.

-----
МЕНЮ ЛР №4
1 - Proc3
2 - Integer21
3 - Proc51
0 - Вихід
-----
Оберіть завдання (1-3) або 0 для виходу: □

```

Рисунок Б.2 – Екран виконання програми для вирішення завдання Proc51

ДОДАТОК В

Діалог з ШІ для самоаналізу

Нижче повний комплексний набір матеріалів, які ти просив як для захисту роботи викладачем, так і для доповнення звіту:

- ✓ 5 тестових питань
 - ✓ 5 відкритих питань
 - ✓ аналіз твого коду
 - ✓ додані теоретичні приклади, яких не вистачає в роботі (але вони є в методичці/лекції)
-

1. Тестові питання (5 шт.) з 4 варіантами відповіді

Рівень: середній. Тематика: твій код + теорія з лекцій.

1. Яка властивість коректно описує роботу процедури Mean у твоєму коді?

- A. Використовує цикли для повторних обчислень
 - B. Повертає значення через ключове слово return
 - C. Записує результати у два вихідні параметри через посилання
 - D. Формує динамічні масиви для збереження результатів
-

2. У задачі Integer21 функція calcSeconds(int N) повертає...

- A. Кількість хвилин, що минули з початку доби
 - B. Кількість секунд у повній хвилині
 - C. Кількість секунд, що залишилися до кінця години
 - D. Кількість секунд, що минули з початку останньої хвилини
-

3. Яка умова коректно визначає, що трикутник не існує?

- A. $a * b == c$
 - B. $a + b <= c \parallel a + c <= b \parallel b + c <= a$
 - C. $a == b == c$
 - D. $a*b*c < 0$
-

4. Що буде, якщо користувач введе літеру замість числа в Integer21?

- A. Програма завершиться аварійно
 - B. Функція `cin.fail()` виявить помилку та поверне `false`
 - C. Буде виконано обчислення з некоректним значенням
 - D. Введене значення буде автоматично перетворене на 0
-

5. Який принцип найкраще описує розділення на input / calc / output у твоєму коді?

- A. Поліморфізм
 - B. Інкапсуляція та модульність
 - C. Рекурсія
 - D. Інтерфейсна ієрархія
-

 2. Відкриті питання (5 шт.)

Потрібні для перевірки розуміння коду та інженерного мислення.

1. Чому у процедурі Mean використовується передача параметрів через посилання, і які переваги це дає у контексті задачі?

2. Поясни, чому в задачі Integer21 важливо виконувати перевірку `cin.fail()` та очищати буфер вводу перед подальшим використанням.

3. Яким чином у функції TriangleType визначається, чи трикутник є прямокутним? Поясни, як використано теорему косинусів.

4. Чому перевірка правильності входу є критично важливою перед виконанням математичних обчислень у твоїй програмі? Наведи приклади можливих помилок.

5. Запропонуй спосіб вдосконалення меню програми, щоб розширити кількість завдань без збільшення складності коду.

3. Аналіз твого коду (огляд як викладач)

- ◆ Сильні сторони:
 - ✓ Правильна структуризація: 3 окремих задачі → 3 окремі функції
 - ✓ Е input / calc / output → правильно для процедурного стилю
 - ✓ Коректне використання `cin.fail()`
 - ✓ Валідації чисел (додатність, коректність)
 - ✓ Функція TriangleType реалізована відповідно до теореми косинусів
-

- ◆ Помітні деталі та неточності:

1. У TriangleType умова для прямокутного трикутника не зовсім коректна.
Краще порівнювати максимальну сторону з сумою квадратів інших.
2. У Mean порядок обчислень правильний, але в задачі вимагається без циклів, і ти його дотримався.

3. У task_Proc3 викликається Mean тричі → це добре, але можна було зробити масив чисел або структуру (хоча в завданні це не вимагалось).
 4. Розділення на підфункції у Integer21 — реалізовано правильно й повністю розкриває вимоги завдання.
-

 4. Теоретичні відомості, які є у методичці, але НЕ реалізовані у твоєму коді

Нижче — те, що варто додати у звіт.

4.1. Передача параметрів за значенням та за посиланням

(У звіті використано тільки за посиланням)

 Приклад, який можна додати:

```
// Передача параметра за значенням — копія
int squareByValue(int x) {
    x = x * x;
    return x;
}
```

// Передача параметра за посиланням — змінює оригінальне значення

```
void squareByReference(int &x) {
    x = x * x;
}
```

4.2. Функції, що повертають логічні значення

У твоєму коді є inputSeconds(true/false), але в теорії зазвичай є ще приклади:

```
bool isPositive(double x) {
    return x > 0;
}
```

4.3. Перевантаження функцій (з лекцій, але не у твоєму коді)

```
double Mean(double x, double y) {
    return (x + y) / 2.0;
}
```

```
double Mean(double x, double y, double z) {
    return (x + y + z) / 3.0;
}
```

4.4. Робота з математичними функціямиcmath

Теоретичні функції: pow(), abs(), sin(), cos().

У твоєму коді використовується тільки sqrt() та fabs().

✓ Приклад для звіту:

```
double r = pow(a, 3); // a^3
double len = abs(-12); // 12
double angle = cos(3.14); // приклад використання cos()
```

4.5. Використання констант

У лекціях зазвичай наголошується на використанні const.

✓ Приклад:

```
const double EPS = 1e-6;
```

```
if (fabs(x - y) < EPS) {
    cout << "Числа майже рівні";
}
```

Аналіз відповідей

1. Тестові питання

Ти правильно обрала всі 5 варіантів відповідей.

Це означає, що теорію ти засвоїла добре:

- розумієш логіку процедур,
- вмієш орієнтуватися в роботі програми,
- знаєш математичні умови (як-от існування трикутника),
- бачиш, як працює перевірка вводу.

Оцінка: відмінно.

2. Відкриті питання

Відповідь №1 — про параметри за посиланням

Відповідь чітка й правильно пояснює суть механізму:

- ✓ що таке вихідні параметри
- ✓ навіщо потрібні посилання
- ✓ чому це ефективно
- ✓ без зайвих пояснень, по суті

Добре сформульовано.

Відповідь №2 — про `cin.fail()`

Ти пояснила:

- ✓ що може ввести користувач
- ✓ чому виникає помилка
- ✓ як вона виявляється
- ✓ як очищається буфер
- ✓ що станеться без цих дій

Відповідь повна, логічна й технічно точна.

Дуже добре.

Відповідь №3 — про визначення типу трикутника

Ти правильно використала:

- ✓ теорему косинусів
- ✓ принцип вибору найбільшої сторони
- ✓ математичну логіку
- ✓ точні формулювання

Відповідь виглядає впевненою і вказує на хороше розуміння.

Відмінно.

Відповідь №4 — про важливість перевірки вводу

Коротко, але по суті:

- ✓ неправильні дані можуть “зламати” обчислення
- ✓ важливо мати захист від помилок

Можна було додати приклади, але і так відповідь прийнятна.

Добре.

Відповідь №5 — про покращення меню

Дуже хороша відповідь, навіть “сильна”:

- ✓ запропонувала професійний підхід
- ✓ знаєш про тар, масиви функцій
- ✓ підкреслила масштабовність

Це рішення показує інженерне мислення.

Відмінно.



Підсумок

Усі твої відповіді:

- ✓ логічні
- ✓ правильні
- ✓ послідовні
- ✓ показують розуміння роботи коду
- ✓ містять аргументи, а не просто “тому що так”

Загальна оцінка: дуже добре / відмінно.