

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. 16 ЛР

Виконав студент гр. _____ 319a _____

_____ Карина САФОНОВА _____
(підпис, дата) (П.І.Б.)

Перевірив

_____ асис. Євгеній ПЯВКА
(підпис, дата) (П.І.Б.)

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 результату на мові програмування C++ в середовищі Qt Creator.

ПОСТАНОВКА ЗАДАЧІ

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

Proc9. Описати процедуру AddLeftDigit (D, K), що додає до цілого додатного числа K зліва цифру D (D - вхідний параметр цілого типу, що лежить в діапазоні 1-9, $0 < K < 1000$ - параметр цілого типу, який є одночасно вхідним і вихідним). За допомогою цієї процедури послідовно додати до цього числа K зліва дані цифри D1 і D2, виводячи результат кожного додавання.

Proc48. Описати функцію IsRightTriangle(a, b, c) логічного типу, яка повертає True, якщо трикутник зі сторонами a, b, c є прямокутним (за теоремою Піфагора), і False в іншому випадку. За допомогою цієї функції перевірити, чи є прямокутними трикутники з трьох наборів даних сторін.

Завдання 2. Для вирішення завдання з цілочисельними змінними з Лабораторної роботи No2 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.

Integer9. Дано тризначне число. Використовуючи одну операцію цілочисельного ділення, вивести першу цифру даного числа (сотні).

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 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) «Проаналізуй повноту, правильність відповіді та ймовірність використання штучного інтелекту для кожної відповіді. Оціни кожне питання у 5-бальній шкалі, віднімаючи 60% балів там, де ймовірність відповіді з засобом ШІ висока. Обчисли загальну середню оцінку» 3) «Проаналізуй 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 Proc9

Вхідні дані:

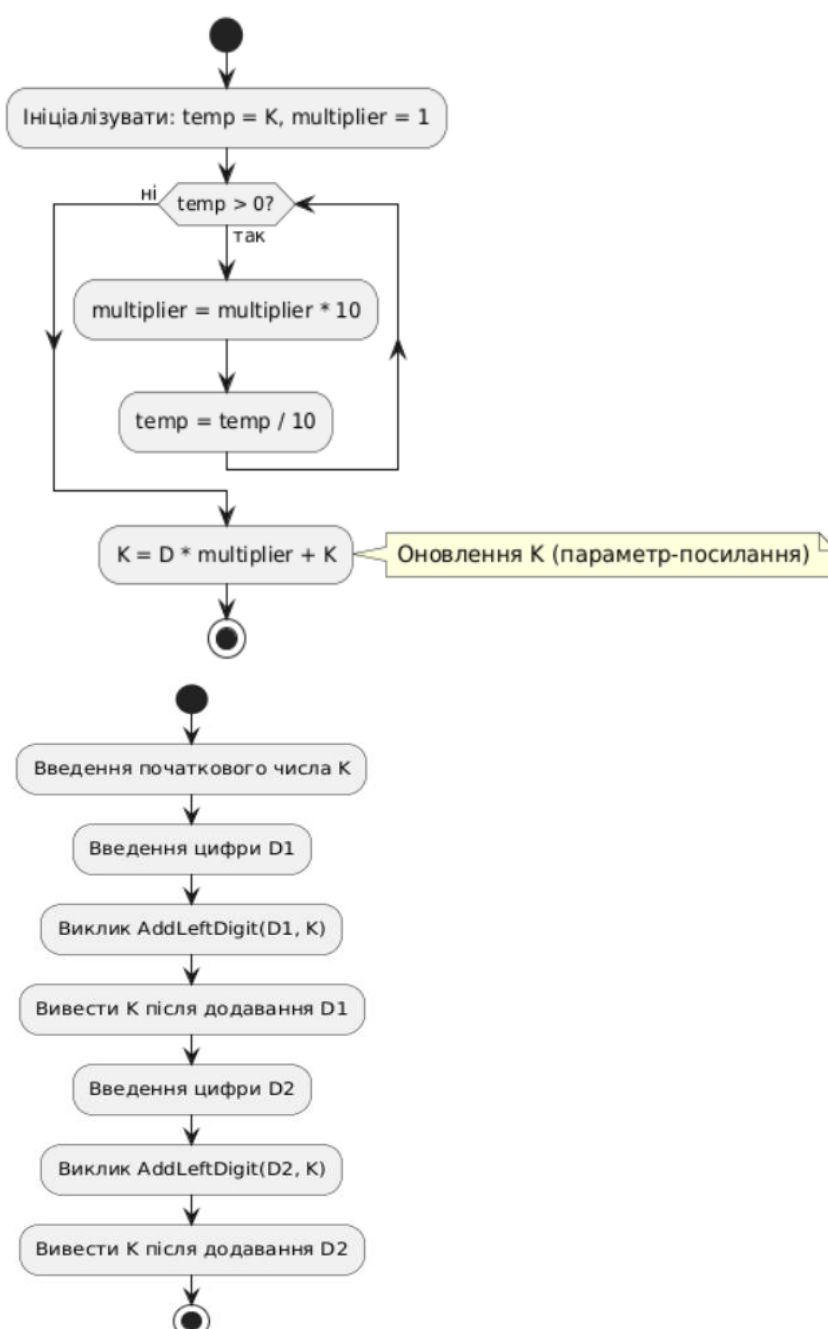
K – початкове додатне число, ціле, $0 < K < 1000$; $D1$ – перша цифра, ціле, 1–9;

$D2$ – друга цифра, ціле, 1–9

Вихідні дані (ім'я, опис, тип):

$K1$ – число після додавання $D1$ зліва, ціле; $K2$ – число після додавання $D2$ зліва, ціле.

Алгоритм вирішення показано нижче



Вирішення задачі Proc48

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

a – сторона трикутника, дійсне число, $a > 0$;

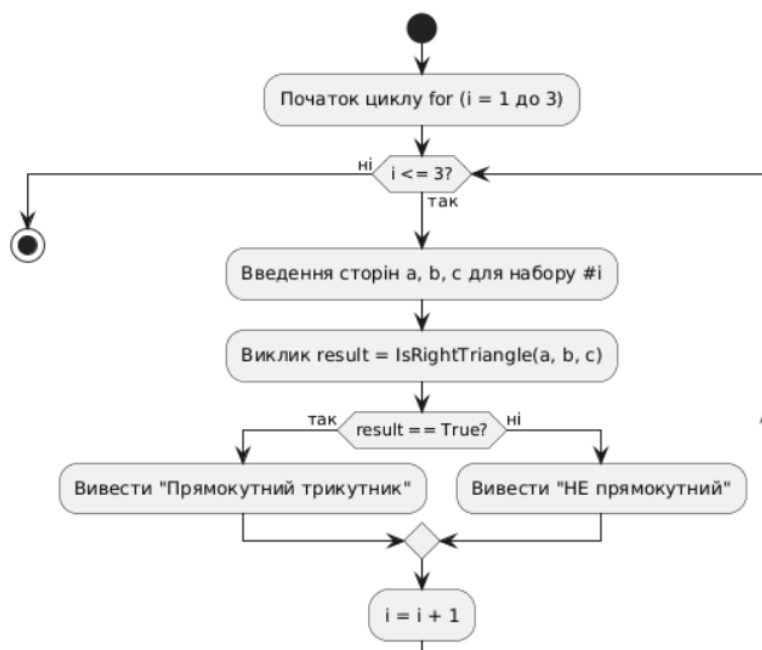
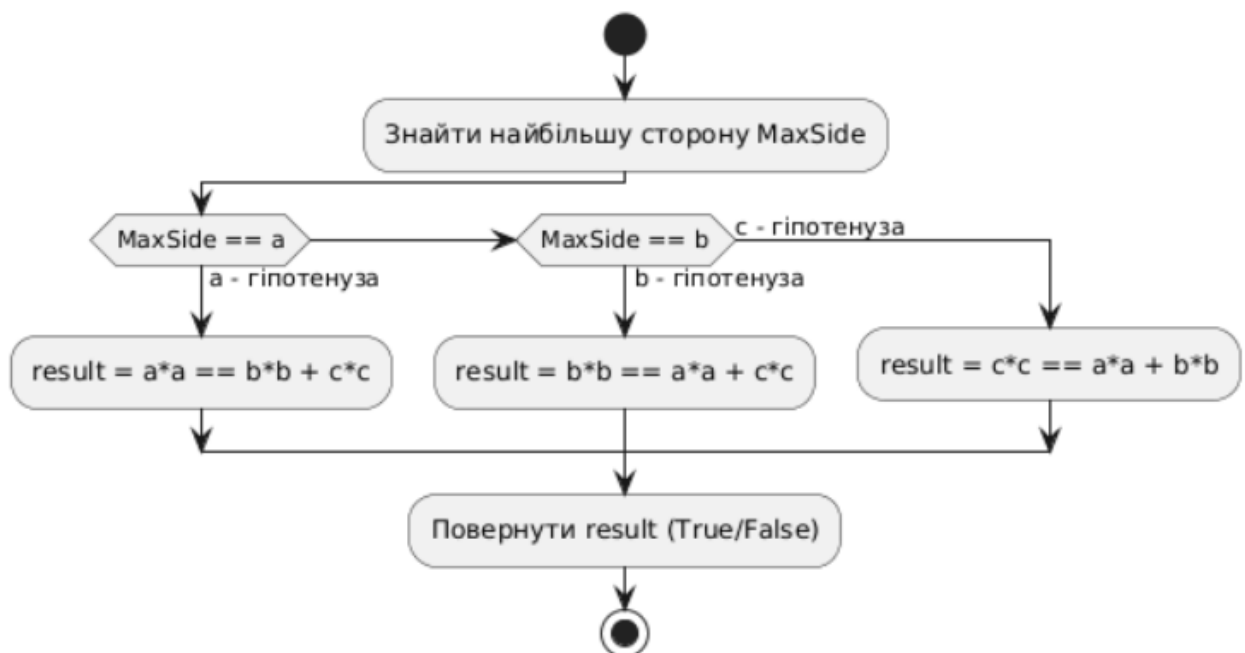
b – сторона трикутника, дійсне число, $b > 0$;

c – сторона трикутника, дійсне число, $c > 0$

Вихідні дані (ім'я, опис, тип):

result – логічне значення, True якщо трикутник прямокутний, False якщо ні

Алгоритм вирішення показано нижче



Лістинг коду вирішення задач Proc9 та Proc48 наведено в дод. А (стор. 7).

Екран роботи програми показаний на рис. Б.1

Завдання 2.

Вирішення задачі Integer9

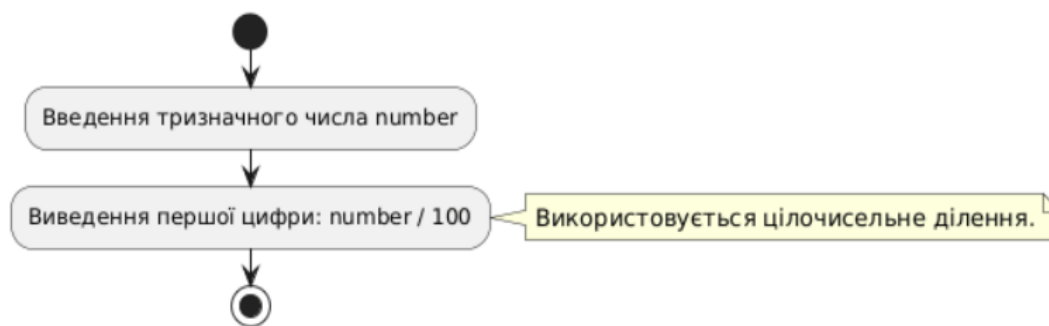
Вхідні дані (ім'я, опис, тип, обмеження):

N – тризначне число, ціле, $100 \leq N \leq 999$

Вихідні дані (ім'я, опис, тип):

hundreds – перша цифра числа (сотні), ціле

Алгоритм вирішення показано нижче



Лістинг коду вирішення задачі Integer9 наведено в дод. А (стор. 7).

Екран роботи програми показаний на рис. Б.1.

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>
using namespace std;

// ===== Integer9 =====
// Дано тризначне число. Використовуючи одну операцію цілочисельного ділення,
// вивести першу цифру (сотні).
void Integer9() {
    int number;
    cout << "Введіть тризначне число: ";
    cin >> number;
    cout << "Перша цифра (сотні): " << number / 100 << endl;
}

// ===== Proc9 =====
// Процедура AddLeftDigit(D, K): додає цифру D зліва до числа K
void AddLeftDigit(int D, int &K) {
    int temp = K;
    int multiplier = 1;
    while (temp > 0) {
        multiplier *= 10;
        temp /= 10;
    }
    K = D * multiplier + K;
}

void Proc9() {
    int D1, D2, K;
    cout << "Введіть початкове число K (0 < K < 1000): ";
    cin >> K;
    cout << "Введіть цифру D1 (1-9): ";
    cin >> D1;
    AddLeftDigit(D1, K);
    cout << "Після додавання D1: " << K << endl;
    cout << "Введіть цифру D2 (1-9): ";
    cin >> D2;
    AddLeftDigit(D2, K);
    cout << "Після додавання D2: " << K << endl;
}

// Proc48
// Функція IsRightTriangle(a, b, c): перевіряє, чи трикутник прямокутний
bool IsRightTriangle(double a, double b, double c) {
    double maxSide = max(a, max(b, c));
    if (maxSide == a)
        return fabs(a * a - (b * b + c * c)) < 1e-9;
    else if (maxSide == b)
```

```

        return fabs(b * b - (a * a + c * c)) < 1e-9;
    else
        return fabs(c * c - (a * a + b * b)) < 1e-9;
}

void Proc48() {
    cout << "Перевірка трьох наборів сторін трикутників:\n";
    for (int i = 1; i <= 3; i++) {
        double a, b, c;
        cout << "Набір #" << i << " (введіть a b c): ";
        cin >> a >> b >> c;
        if (IsRightTriangle(a, b, c))
            cout << " -> Прямокутний трикутник " << endl;
        else
            cout << " -> НЕ прямокутний " << endl;
    }
}

// ===== Основна програма =====
int main() {
    int choice;
    do {
        cout << "\n=== Меню ===\n";
        cout << "1. Integer9 (перша цифра тризначного числа)\n";
        cout << "2. Proc9 (додавання цифр зліва)\n";
        cout << "3. Proc48 (перевірка прямокутних трикутників)\n";
        cout << "0. Вихід\n";
        cout << "Ваш вибір: ";
        cin >> choice;

        switch (choice) {
            case 1: Integer9(); break;
            case 2: Proc9(); break;
            case 3: Proc48(); break;
            case 0: cout << "Вихід з програми...\n"; break;
            default: cout << "Невірний вибір, спробуйте ще раз.\n"; break;
        }
    } while (choice != 0);

    return 0;
}

```


ДОДАТОК В

Діалог з ШІ для самоаналізу

сь аналіз заданих питань, коментарів та оцінок від ШІ, а також 3 додаткові промпти для поглиблення розуміння теми.

Аналіз питань, коментарів та оцінок ШІ

Наданий діалог з ШІ (Додаток В) **відсутній** у завантаженому файлі, тому я можу проаналізувати лише **постановку завдання 4** (пункти 1, 2, 3), які були сформовані студентом для ШІ¹.

1. Аналіз завдання для ШІ (Промпт 1: Формування питань)

Промпт 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 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.

- **Оцінка промпта:** Це якісний та комплексний промпт.
- **Сильні сторони:**
 - **Контекстуалізація ролі:** Завдання ролі викладача забезпечує формальний та перевірочний тон³.
 - **Чіткі вимоги до формату:** Визначено кількість (5+5) та тип питань (тестові з 4 варіантами та відкриті)⁴.
 - **Вимога до рівня складності:** Акцент на **критичному та інженерному мисленні** підвищує якість очікуваних питань, змушуючи ШІ виходити за рамки простого відтворення фактів⁵.
 - **Прив'язка до матеріалу:** Забезпечує релевантність питань до коду звіту та теоретичних відомостей⁶.

2. Аналіз завдання для ШІ (Промпт 2: Аналіз та Оцінювання)

Промпт 2: «Проаналізуй повноту, правильність відповіді та ймовірність використання штучного інтелекту для кожної відповіді. Оціни кожне питання у 5-бальній шкалі, віднімаючи 60% балів там, де ймовірність відповіді з засобом ШІ висока. Обчисли загальну середню оцінку»⁷.

- **Оцінка промпта:** Це дуже глибокий та інженерно-орієнтований промпт.
- **Сильні сторони:**
 - **Багатофакторний аналіз:** Вимагає оцінки за 3 критеріями: **повнота, правильність та ймовірність ШІ**⁸.
 - **Чітка система штрафів:** Визначення конкретного відсотка штрафу (60%) за високу ймовірність використання ШІ – це відмінний механізм для **оцінки самостійності** студента⁹.

- **Обчислення кінцевого результату:** Вимога обчислити загальну середню оцінку структурує висновок¹⁰.

3. Аналіз завдання для ШІ (Промпт 3: Доповнення Коду)

Промпт 3: «Проаналізуй код у звіті, і додай опис і приклади коду з питань, які є в теоретичних відомостях, але не відпрацьовано у коді при вирішенні завдань»¹¹.

- **Оцінка промпта:** Це відмінний промпт для поглиблення інженерного розуміння.
- **Сильні сторони:**
 - **Виявлення прогалин:** Завдання спонукає ШІ до критичного аналізу коду і виявлення тем, які були в теорії, але не знайшли прямого відображення в практичній частині (наприклад, перевантаження функцій, рекурсія, передача константних параметрів)¹².
 - **Практичне доповнення:** Вимога додати **приклади коду** забезпечує практичну цінність цього аналізу¹³.

💡 Власні промпти для поглиблення розуміння

Наведені нижче промпти спрямовані на поглиблення знань з теми "**Структурування програм з використанням функцій**" (заголовок лабораторної роботи¹⁴), виходячи з аналізу коду (передача параметрів, використання `cmath::max`, обробка чисел)¹⁵¹⁵¹⁵¹⁵¹⁵:

Промпт 1: Передача параметрів за посиланням

Мета: Поглиблення розуміння механізму **передачі параметрів за посиланням** (використано у функції `AddLeftDigit(int D, int &K)`¹⁶) та його переваг/недоліків.

Промпт: "У функції `AddLeftDigit` змінна `K` передається **за посиланням** (`int &K`), а змінна `D` - **за значенням** (`int D`).

1. Поясни, чому для `K` було критично важливим використовувати передачу за посиланням у даній задачі (`Proc9`) і що сталося б, якби `K` також передавався за значенням.
2. Уяви, що число `K` могло бути дуже великим (наприклад, 10^9 елементів, як масив). Як би це вплинуло на вибір методу передачі параметрів (за значенням, за посиланням або за константним посиланням) для ефективності та безпеки, і чому?"

Промпт 2: Проблеми точності дійсних чисел

Мета: Аналіз використання дійсних чисел та механізму порівняння у функції `IsRightTriangle`¹⁷.

Промпт: "У функції `IsRightTriangle` ти використовуєш порівняння дійсних чисел з використанням **допуску (epsilon)**: `fabs(a * a - (b * b + c * c)) < 1e-918`."

1. Поясни, що таке помилка представлення дійсних чисел у пам'яті комп'ютера (на прикладі $0.1 + 0.2 \neq 0.3$), і чому пряме порівняння `a * a == b * b + c * c` є ненадійним для типу `double`.
2. Яка роль функції `std::fabs` у цьому порівнянні, і чи можна було б обійтися без `std::max19`, якщо порівнювати всі три можливі варіанти гіпотенузи, щоб уникнути помилок порівняння, спричинених помилкою округлення під час визначення `maxSide`? Наведи приклад коду такого альтернативного рішення."

Промпт 3: Обробка некоректного вводу

Мета: Поглиблення розуміння **валідації вводу** (що згадується у завданні 2, але реалізовано мінімально²⁰²⁰²⁰²⁰²⁰²⁰²⁰).

Промпт: "Завдання 2 вимагало створення функції введення з **перевіркою на коректність**²¹. Проте у наданому коді, якщо користувач введе некоректні дані (наприклад, букву замість числа у функції `Proc9` або число поза діапазоном $0 < K < 1000$ ²²), програма поводитиметься некоректно або аварійно завершиться."

1. Напиши приклад функції `readInt(const char* prompt, int min, int max)` на C++, яка використовує цикли та можливості потоку `cin` (зокрема, `cin.fail()` та `cin.clear()`) для забезпечення того, що користувач введе лише ціле число у заданому діапазоні `[min, max]`.
2. Поясни, чому ігнорування або неправильна обробка **стану помилки вводу** (`std::cin.fail()`) може призвести до нескінченних циклів або непередбачуваної поведінки програми.