

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний аерокосмічний університет

«Харківський авіаційний інститут»

Факультет систем управління літальних апаратів

Кафедра систем управління літальних апаратів

Лабораторна робота № 3

з дисципліни «Алгоритмізація та програмування»

на тему «"Реалізація алгоритмів з розгалуженням мовою C ++"»

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.)

2025

МЕТА РОБОТИ

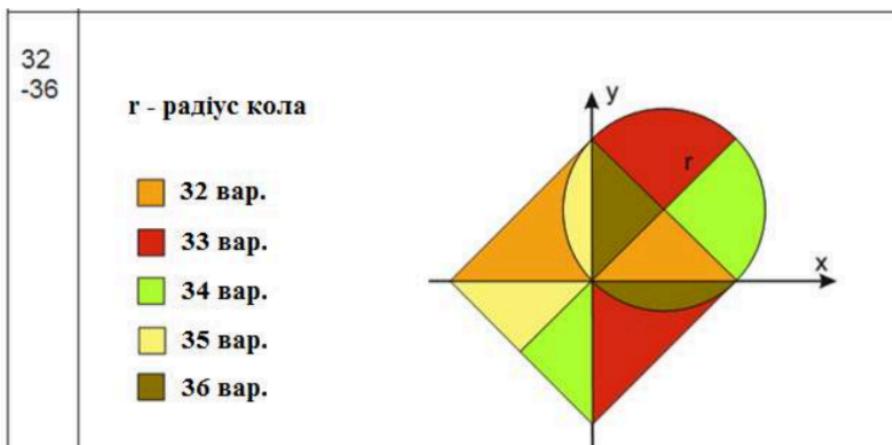
Вивчити теоретичний матеріал щодо синтаксису у мові С ++ і подання у вигляді UML діаграм активності алгоритмів з розгалуженням та реалізувати алгоритми з використанням інструкцій умовного переходу і вибору мовою С++ в середовищі QtCreator. Також опанувати та відпрацювати навички структурування програми з функціями.

ПОСТАНОВКА ЗАДАЧІ

Завдання 1. Вирішити дві задачі на алгоритми з розгалуженням.

If9. Дано дві змінні дійсного типу: А, В. Перерозподілити значення даних змінних так, щоб в А було записано менше зі значень, а в В – більше. Вивести нові значення змінних А і В.

Завдання 2. Дано координати точки на площині (x, y). Визначити, чи потрапляє точка в фігуру заданого кольору (або групу фігур) і вивести відповідне повідомлення.



32 вар.

Завдання 3. Для вибору користувачем одного з трьох зазначених вище завдань розробити алгоритм організації меню в командному вікні з використанням інструкції вибору.

Завдання 4. Використовуючи ChatGpt, Gemini або інший засіб генеративного ІІ, 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) «Проаналізуй 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 Ш висока. Обчисли загальну середню оцінку» Проаналізуйте задані питання, коментарі і оцінки, надані Ш. Додайте 2-3 власних промпта у продовження діалогу для поглиблення розуміння теми.

ВИКОНАННЯ РОБОТИ

Завдання 1.

Вирішення задачі If9

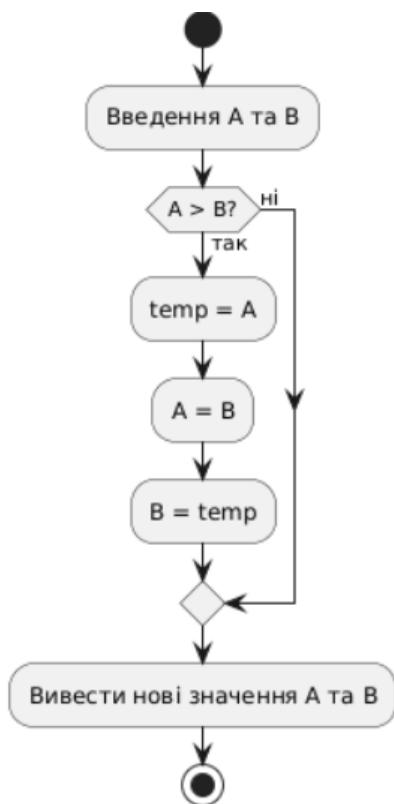
Вхідні дані (ім'я, опис, тип, обмеження):

- А – перша змінна, дійсного типу, double, будь-яке дійсне число
- В – друга змінна, дійсного типу, double, будь-яке дійсне число

Вихідні дані (ім'я, опис, тип):

- А – менше з початкових значень, double
 - В – більше з початкових значень, double
- текст – повідомлення користувачу про введення даних або результат, string

Алгоритм вирішення показано нижче:



1. Вивести заголовок програми.
2. Ввести значення змінних A і B (дійсні числа).
3. Перевірити правильність введення даних.
4. Виконати розгалуження:
якщо $A > B$, то поміняти їх місцями;
інакше залишити без змін.
5. Вивести нові значення змінних A і B ($A \leq B$).

Екран роботи програми показаний на рис. Б.1.

Завдання 2.

Вирішення задачі Figure 32

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

x – координата точки по осі X, double, будь-яке дійсне число
 y – координата точки по осі Y, double, будь-яке дійсне число
 r – радіус кола, double, $r > 0$

Вихідні дані (ім'я, опис, тип):

"Точка в ЗЕЛЕНИЙ області (І чверть)." – точка потрапляє у зелену частину кола,

текст

"Точка в ЖОВТІЙ області (II чверть)." – точка потрапляє у жовту частину кола, текст

"Точка в ЧЕРВОНІЙ області (III чверть)." – точка потрапляє у червону частину кола, текст

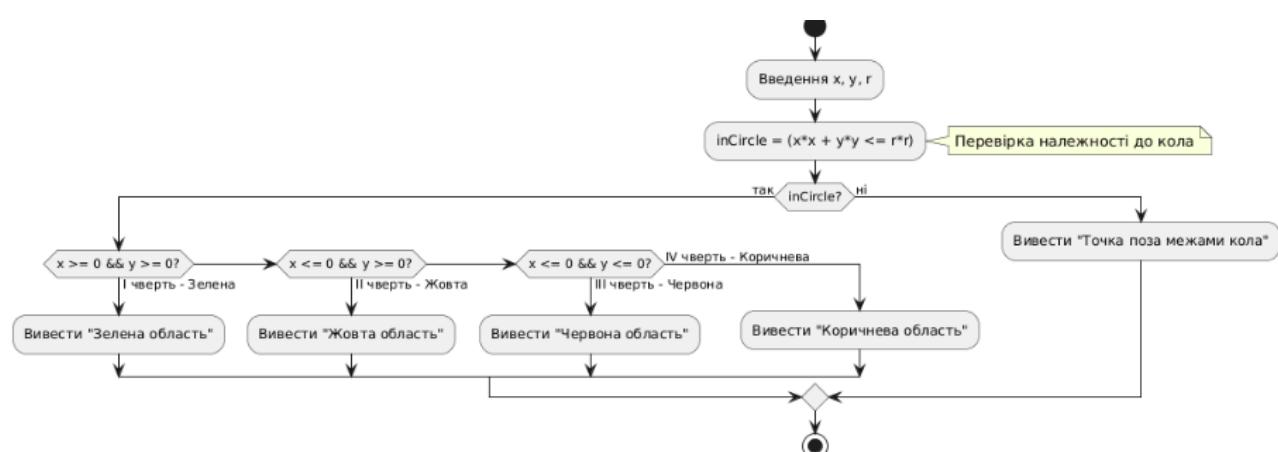
"Точка в КОРИЧНЕВІЙ області (IV чверть)." – точка потрапляє у коричневу частину кола, текст

"Точка поза колом." – точка не належить колу, текст

"Помилкове значення для $r!$ " – введено недопустиме значення радіуса, текст

"Некоректні координати!" – введено некоректні координати точки, текст

Алгоритм вирішення показано нижче:



1. Зчитати значення r (радіус кола).
2. Якщо $r \leq 0$ – вивести повідомлення "Помилкове значення для $r!$ " і завершити роботу.
3. Зчитати координати точки (x, y) .
4. Перевірити правильність введення координат.
5. Перевірити, чи належить точка колу за умовою: $x^2 + y^2 \leq r^2$.
Якщо точка не належить колу – вивести повідомлення "Точка поза колом".
6. Якщо точка належить колу, визначити її положення:
якщо $x \geq 0, y \geq 0$ – точка у зеленій області (I чверть);
якщо $x \leq 0, y \geq 0$ – точка у жовтій області (II чверть);
якщо $x \leq 0, y \leq 0$ – точка у червоній області (III чверть);
якщо $x \geq 0, y \leq 0$ – точка у коричневій області (IV чверть).

7. Вивести повідомлення про належність точки до відповідної кольорової області.

Екран роботи програми показаний на рис. Б.2

ВИСНОВКИ

Було опрацьовано принципи побудови алгоритмів із розгалуженням та їх реалізацію в середовищі програмування C++. На практиці закріплено навички створення меню вибору завдань і перевірки умов належності точки заданий фігурі. Відпрацьовано використання логічних виразів, операторів розгалуження та обробку введення користувача. Набуто вміння будувати UML-діаграми активності для візуалізації структури програми. Певні труднощі виникли під час визначення меж фігури та формулювання логічних умов перевірки.

ДОДАТОК А

Лістинг коду програми

```

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

int main() {
    cout << " if 9: ";
setlocale(LC_ALL, "ukr");
    //Завдання 1
    double A, B;
    cout << "Введіть A і B: ";
    cin >> A >> B;

    if (A > B) {
        double temp = A;
        A = B;
        B = temp;
    }

    cout << "Менше значення (A): " << A << endl;
    cout << "Більше значення (B): " << B << endl;

    //Завдання 2
    cout << " figure 32: ";
    double x, y, r;
    cout << "\nВведіть координати точки (x, y): ";
    cin >> x >> y;
    cout << "Введіть радіус кола r: ";
    cin >> r;

    // умова належності до кола
    bool inCircle = (x * x + y * y <= r * r);

    if (!inCircle) {
        cout << "Точка поза межами кола." << endl;
    } else {
        if (x >= 0 && y >= 0) {
            cout << "Точка у зеленій області (I чверть)." << endl;
        } else if (x <= 0 && y >= 0) {
            cout << "Точка у жовтій області (II чверть)." << endl;
        } else if (x <= 0 && y <= 0) {
            cout << "Точка у червоній області (III чверть)." << endl;
        } else if (x >= 0 && y <= 0) {
            cout << "Точка у коричневій області (IV чверть)." << endl;
        }
    }
}

return 0;

```

}

ДОДАТОК Б
Скрін-шоти вікна виконання програми

```
if 9: Введіть A і B: 55 77
Менше значення (A) : 55
Більше значення (B) : 77
```

Рисунок Б.1 – Екран виконання програми для вирішення завдання
IF 9

```
figure 32:
Введіть координати точки (x, y): 55433
8776
Введіть радіус кола r: 363758779
Точка у зеленій області (I чверть).
```

Рисунок Б.2 – Екран виконання програми для вирішення завдання
Figure 32

ДОДАТОК В

Діалог з ШІ для самоаналізу

Це чудове завдання для самоаналізу! Ось 2-3 додаткових промпта (підказок) для продовження діалогу з генеративним ШІ, спрямованих на поглиблення розуміння теми лабораторної роботи та коду.

Додаткові Промпти для Діалогу з ШІ

Ці промпти допоможуть розвинути критичне та інженерне мислення, а також детальніше проаналізувати написаний код і теоретичні основи.

Промпт 1: Оптимізація та Альтернативні Підходи

Промпт: "Проаналізуй мій код для Завдання 1 (lf9). Запропонуй два альтернативних способи вирішення задачі перерозподілу значень змінних `A` і `B`, один з яких повинен використовувати стандартну функцію (якщо така існує), а інший — покажчики або посилання (якщо вони доречні для цієї задачі). Порівняй ефективність, читабельність та потенційні ризики кожного з трьох підходів (мій, з функцією, з покажчиками/посиланнями)."

- Мета: Розвиток інженерного мислення, розуміння ефективності коду, використання стандартних бібліотек (`std::swap`, `std::min/std::max`) та механізмів передачі даних (посилання/покажчики).
-

Промпт 2: Вдосконалення Логіки та Обробка Помилок

Промпт: "У Завданні 2 (Figure 32) я використав просту умову $x^2 + y^2 \leq r^2$ для перевірки належності точки колу. Моя реалізація алгоритму в розділі 'Виконання роботи' передбачала перевірку на `$r \leq 0$` та 'некоректні координати', але в коді (Додаток А) я не реалізував явну перевірку на `$r \leq 0$` та помилки введення.

1. Які потенційні проблеми можуть виникнути в моїй поточній реалізації, якщо користувач введе некоректні дані (наприклад, $r < 0\$$ або нечислові значення для $x, y, r\$$)?
 2. Запропонуй концепцію функції (прототип та коротке описання логіки), яка б виконувала надійну перевірку вхідних даних (включаючи $r > 0\$$ та успішність `cin`) і повертала б відповідний статус помилки або коректні значення."
- Мета: Розвиток критичного мислення, навичок обробки помилок та захищеного програмування, а також розуміння різниці між алгоритмом та його повною реалізацією.

Промпт 3: Узагальнення та Архітектура Програми

Промпт: "Для Завдання 3 мені було необхідно реалізувати меню за допомогою інструкції вибору (switch або вкладених if-else). Уявімо, що наша програма розширюється, і тепер в ній буде не 3, а 15 різних завдань (функцій).

1. Як архітектурно доцільно змінити структуру програми `main()` для більш гнучкого та масштабованого керування вибором завдань?
 2. Яку інструкцію розгалуження (або інший механізм) слід використовувати для реалізації меню з 15 пунктами, щоб код був найбільш чистим та легким для підтримки?
Обґрунтуй свій вибір, спираючись на принципи чистого коду."
- Мета: Розвиток інженерного мислення та навичок архітектури програми, розуміння масштабованості та застосування принципів чистого коду при роботі з розгалуженнями.