

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. 31 ЛР

Виконав студент гр. 319a

Максим САВЧИН

(підпис, дата)

(П.І.Б.)

Перевірив

асис. Євгеній ПЯВКА

(підпис, дата)

(П.І.Б.)

## МЕТА РОБОТИ

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

## ПОСТАНОВКА ЗАДАЧІ

Завдання 1. Вирішити дві задачі на алгоритми з розгалуженням.

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

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 ШІ, провести самоаналіз отриманих знань і навичок за допомогою наступних промптів:

1) «*Tи - 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»*

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

### ВИКОНАННЯ РОБОТИ

#### Завдання 1.

Вирішення задачі if11. Дано дві змінні цілого типу: А і В. Якщо їх значення не рівні, то присвоїти кожній змінній більше з цих значень, а якщо рівні, то присвоїти змінним нульові значення. Вивести нові значення змінних А і В.

Вхідні дані: А і В – значення змінних.

Вихідні дані: А і В – нові значення змінних.

Алгоритм вирішення:

- 1) Введення значень А і В;
- 2) Якщо  $A = B$ , то присвоїти обом змінним значення 0;
- 3) Якщо  $A \neq B$ , то присвоїти змінним більше із цих значень;
- 4) Виведення нових значень А і В.

Лістинг коду вирішення задачі if11 наведено в дод. А

Екрани роботи програми показано на рис. Б.1

#### Завдання 2.

Вирішення завдання Geom30.

Вхідні дані: r – радіус, x і y – координати точки.

Вихідні дані: положення точки.

Алгоритм вирішення:

- 1) Введення значення радіуса  $r$  та координат точки ( $x, y$ );
- 2) Перевірка коректності введених даних;
- 3) Обчислити де знаходиться точка;
- 4) Вивести положення точки.

Лістинг коду вирішення задачі geom30 наведено в дод. А

Екрани роботи програми показано на рис. Б.2

#### Завдання 3.

Лістинг алгоритму організації меню наведено в дод. А

## Завдання 4.

Діалог з ШІ для самоаналізу наведено у дод.В.

## ВИСНОВКИ

Було вивчено принцип побудови і подання у вигляді UML діаграм активності алгоритмів з розгалуженням. Опановано синтаксис умовних операторів if-else та інструкції вибору switch. Було опановано і розроблено програму з вибором двох функцій.

## ДОДАТОК А

### Лістинг коду програми

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

//If11. Дано дві змінні цілого типу: A і B. Якщо їх значення не рівні,
//то присвоїти кожній змінній більше з цих значень, а якщо рівні,
//то присвоїти змінним нульові значення.
//Вивести нові значення змінних A і B.
void task_if11 (); // завдання 1 декларація функції

// Дано координати точки на площині (x, y).
// Визначити, чи потрапляє точка в фігуру заданого кольору (або групу фігур)
// і вивести відповідне повідомлення
void task_geom30 (); //завдання 2 декларація функції

int main() {
    int menu;
    cout << "Task number:";
    cin >> menu;
    // перемикання між завданнями
    switch (menu)
    {
        case 1: task_if11 (); break; // Завдання 1
        case 2: task_geom30 (); break; // Завдання 2
        default: cout << "Wrong task! (Only 1,2)" << endl; // повідомлення про
помилку
    }
    return 0;
}

// завдання 1 реалізація
void task_if11 ()
{
    int A, B;
    cout << " If11 " << endl;
    cout << " Введіть значення змінної A: ";
    cin >> A;

    if (cin.fail()) {
        cout << "Некоректні дані!";
        return;
    }

    cout << " Введіть значення змінної B: ";
    cin >> B;

    if (cin.fail()) {
        cout << "Некоректні дані!";
        return;
    }
}
```

```

    if (A == B) {
        cout << "Змінні рівні!";
        A = 0;
        B = 0;
    }
    else {

        A<B ? A=B : B=A;
        cout << "Змінні відрізняються!";
    }

    cout << "A = " << A << "B = " << B << endl;

    return;
}

// завдання 2 реалізація
void task_geom30 () {
    double r; // радіус
    double x, y; // координати точки
    cout << " Geom30 " << endl;
    cout << " Введіть радіус кола: ";
    cin >> r;
    if (cin.fail() || r <= 0) {
        cout << "Некоректне значення радіусу! ";
        return;
    }
    cout << "Введіть координати x, y: ";
    cin >> x >> y;
    if (!cin) {
        cout << "Задано не числові дані! ";
        return;
    }
    float eps = 1e-9;

    bool at_upper_right = (-r < x && x <= 0) && (0 < y && y <= r);
    double dist2 = pow(x + r, 2) + pow(y, 2);
    bool outside_circle = dist2 > pow(r, 2) + eps;

    bool in_triangle = false;
    {
        double Ax = -2 * r, Ay = 0;
        double Bx = -r, By = 0;
        double Cx = -r, Cy = -r;

        double v0x = Bx - Ax, v0y = By - Ay;
        double v1x = Cx - Ax, v1y = Cy - Ay;
        double v2x = x - Ax, v2y = y - Ay;
        double det = v0x * v1y - v1x * v0y;

        if (abs(det) > eps) {
            double u = (v2x * v1y - v1x * v2y) / det;
            double v = (v0x * v2y - v2x * v0y) / det;

```

```
    in_triangle = (u >= -eps) && (v >= -eps) && (u + v <= 1 + eps);
}

if (in_triangle)
    cout << "Точка належить області 30 (нижня фігура, світло-зелений колір)";
else if (at_upper_right && outside_circle)
    cout << "Точка належить області 30 (верхня фігура, світло-зелений
колір)";
else {
    cout << "Задана точка не знаходиться на фігурі!";
}

return;
}
```

## ДОДАТОК Б

### Скрін-шоти вікна виконання програми

The image shows two separate terminal windows side-by-side. Both windows have a title bar with icons and a status bar labeled "input".  
The top window displays the following output:  
Task number:1  
If11  
Введіть значення змінної A: 5  
Введіть значення змінної B: 5  
Змінні рівні!A = 0B = 0  
  
...Program finished with exit code 0  
Press ENTER to exit console.  
  
The bottom window displays the following output:  
If11  
Введіть значення змінної A: 4  
Введіть значення змінної B: 6  
Змінні відрізняються!A = 6B = 6  
  
...Program finished with exit code 0  
Press ENTER to exit console.

рис. Б.1

The image shows three separate terminal windows side-by-side, all with a title bar and "input" status bar.  
The first window (top) displays:  
Task number:2  
Geom30  
Введіть радіус кола: 10  
Введіть координати x, y: 8  
3  
Задана точка не знаходиться на фігури!  
  
...Program finished with exit code 0  
Press ENTER to exit console.  
  
The second window (middle) displays:  
Task number:2  
Geom30  
Введіть радіус кола: 2  
Введіть координати x, y: -2.5 -1  
Точка належить області 30 (нижня фігура, світло-зелений колір)  
  
...Program finished with exit code 0  
Press ENTER to exit console.  
  
The third window (bottom) displays:  
Task number:2  
Geom30  
Введіть радіус кола: 2  
Введіть координати x, y: -0.5 1.5  
Точка належить області 30 (верхня фігура, світло-зелений колір)  
  
...Program finished with exit code 0  
Press ENTER to exit console.

рис.Б.2

## 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. 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

Добре, Максиме

Я – викладач, який приймає у тебе захист лабораторної роботи №3 з дисципліни «Алгоритмізація та програмування».

Нижче наведено 5 тестових запитань із варіантами відповідей і 5 відкритих запитань.

Вони стосуються коду, поданого у твоєму звіті (функції task\_if1(), task\_geom30() та main()), і теоретичних знань із лекцій – зокрема, теми розгалуження, інструкцій вибору, перевірки даних і роботи з умовами в C++.

#### **Частина 1. Тестові запитання (обери одну правильну відповідь)**

1. Яка особливість використання інструкції switch у функції main() твоєї програми?

- а) Вона дозволяє порівнювати будь-які логічні вирази.
- б) Вона може обробляти лише цілі або символільні значення.
- в) Вона ефективніше за if-else працює з числами з плаваючою комою.
- г) Вона автоматично перевіряє правильність введення даних.

---

2. У функції task\_if1() використовується вираз  $A < B ? A = B : B = A;$ . Що відбувається після його виконання, якщо змінні не рівні?

- а) Обидві змінні отримують менше із двох значень.
  - б) Обидві змінні отримують більше із двох значень.
  - в) А отримує менше, В – більше значення.
  - г) Обидві змінні обнуляються.
-

3. У функції `task_geom30()` застосовується змінна `eps = 1e-9`. Для чого вона використовується?

- а) Для округлення результатів до 9 знаків після коми.
  - б) Для підвищення точності при порівнянні чисел з плаваючою комою.
  - в) Для прискорення обчислень при перевірці умов.
  - г) Для задання максимально допустимого радіуса кола.
- 

4. Яку функцію виконує блок `if (cin.fail())` у коді програми?

- а) Виводить повідомлення про помилку компіляції.
  - б) Перевіряє, чи було введено дані неправильного типу.
  - в) Визначає, чи введено число, менше за нуль.
  - г) Очищує буфер введення перед новим вводом.
- 

5. У фрагменті коду з `task_geom30()` використовується змінна `dist2 = pow(x + r, 2) + pow(y, 2);`. Яку роль відіграє цей вираз?

- а) Обчислює відстань від точки  $(x, y)$  до початку координат.
  - б) Обчислює квадрат відстані між точкою  $(x, y)$  та центром кола  $(-r, 0)$ .
  - в) Визначає площину кола.
  - г) Визначає напрямок вектора, що сполучає точку з центром.
- 

□

## **Частина 2. Відкриті запитання (поясни своїми словами)**

1. Поясни логіку роботи функції `task_if11()`: як саме програма змінює значення змінних А і В залежно від їх порівняння.

---

2. У функції `task_geom30()` одночасно використовуються перевірки для трикутника і для області за колом. Поясни, як програма визначає, до якої саме області належить точка.

---

3. Які переваги має використання двох окремих функцій (`task_if11()` і `task_geom30()`) замість запису всього коду безпосередньо в `main()`.

---

4. Поясни різницю між логічними операціями `&&`, `||` і `!` у C++. Наведи приклади, коли кожна з них застосовується в умовах розгалуження.

---

5. Що таке діаграма активності UML? Які елементи вона містить і яку роль вона відіграє у проектуванні алгоритмів з розгалуженням?

тестові питання:

- 1) Б
- 2) Б
- 3) Б
- 4) Б
- 5) Б

Відкриті питання:

- 1) Програма перевіряє змінні на рівність, якщо вони рівні то присвоює обом значення 0, в іншому випадку обом присвоюється більше значення.
- 2) Спочатку перевіряється чи належить точка нижній фігури, якщо так то це виводиться на консоль, в іншому випадку перевіряється належність верхній фігури, для цього використовується `if..else`.
- 3) Легше виявляти помилки в коді та на консоль виводиться тільки одна функція, що робить код зручнішим для читання
- 4) В C++ ці оператори використовують лініїві обчислення (short-circuit evaluation). Наприклад, при використанні `&&`, якщо перший операнд є хибним (`false`), другий операнд не обчислюється, оскільки результат вже точно буде `false`. Це підвищує ефективність та запобігає помилкам (наприклад, діленню на нуль).
- 5) Діаграма Активності – це UML-модель, що візуалізує послідовність кроків та потік керування алгоритмом. Діаграма слугує чітким графічним шаблоном для кодування, дозволяючи візуально перевірити (валідувати), чи всі можливі шляхи та умови розгалуження в алгоритмі були враховані та коректно структуровані.

Проаналізуй повноту, правильність відповідіта ймовірність використання штучного інтелекту дляожноївідповіді. Оціни кожне питання у 5-балльній шкалі, віднімаючи 60% балів там, де ймовірність відповіді з засобом ШІ висока. Обчисли загальнусередню оцінку

Усі тестові відповіді є повними, правильними та точними. Студент обрав варіанти Б для всіх п'яти питань, що повністю відповідає правильним відповідям згідно з

теоретичним матеріалом та кодом програми. Формулювання короткі, конкретні, без ознак використання сторонніх джерел. Ймовірність застосування штучного інтелекту при наданні тестових відповідей оцінюється як низька. Середній бал за тестову частину становить 5,0 з 5 можливих.

Відкриті відповіді також продемонстрували розуміння принципів умовних операторів, структуризації програми та логіки перевірки належності точки до області. Відповіді на перші три питання (1-3) є повними, логічно послідовними та відображають реальний зміст програмного коду. Ймовірність використання штучного інтелекту в цих відповідях низька.

У відповідях на питання 4 і 5 (пояснення роботи логічних операторів і UML-діаграми активності) спостерігається високий рівень формальності, академічний стиль викладу і використання типових термінологічних конструкцій, що характерно для відповідей, згенерованих за допомогою моделей штучного інтелекту. У зв'язку з цим для цих двох пунктів оцінку знижено на 60%.

Загальний середній бал за відкриті питання становить 3,7 із 5 можливих.

Узагальнена оцінка за всі питання лабораторної роботи (з урахуванням тестової і відкритої частини) складає 4,35 бала з 5, що відповідає рівню «добре».

**Висновок:** студент продемонстрував достатній рівень знань теоретичного матеріалу, розуміння логіки програмного коду, вміння аналізувати умови та алгоритмічні структури. Ймовірність використання штучного інтелекту у частині відповідей є помірною, але не знижує загальну якість роботи. Загальна оцінка - 4,3 бала з 5.