

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний аерокосмічний університет ім. М. Є. Жуковського
«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. 174. 319. 23 ЛР

Виконав студент гр. 319

Віталій НОВИКОВ

(підпис, дата)

(П.І.Б.)

Перевірів

к.т.н., доц. Олена ГАВРИЛЕНКО

(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 Visual Studio. Також опанувати та відпрацювати навички структурування програми з функціями.

ПОСТАНОВКА ЗАДАЧІ

Завдання 1. Вирішити завдання на алгоритми з розгалуженням.

If30. Дано ціле число, яке лежить в діапазоні 1-999. Вивести його рядок-опис виду «парне двозначне число», «непарне тризначне число» і т. ін..

Завдання 2. Дано координати точки на площині (x, y). Визначити, чи потрапляє точка в фігуру заданого кольору (або групу фігур) і вивести відповідне повідомлення. Фігуру представлено на рис.1



рис.1

Завдання 3. Обчислити площу і периметр плоскої фігури.

Завдання 4. Для вибору користувачем одного з трьох зазначених вище завдань розробити алгоритм організації меню в командному вікні з використанням інструкції вибору.

ВИКОНАННЯ РОБОТИ

Завдання 1. Опис числа

Вирішення задачі: If30

Вхідні дані:

- n: ціле число (тип int), діапазон: [1, 999]

Вихідні дані:

- parity: рядок (тип string), опис парності числа.
- digits: рядок (тип string), опис кількості цифр у числі (однозначне, двозначне, тризначне).

Алгоритм вирішення:

1. Запитати у користувача ввести число n.
2. Перевірити, чи входить число n у діапазон [1, 999].
3. Визначити парність числа (парне або непарне).
4. Визначити кількість цифр у числі (однозначне, двозначне, або тризначне).
5. Вивести результат у вигляді текстового опису.

Екран роботи програми показаний в додатку Б, рис. Б.1(стор.8).

Завдання 2. Перевірка точки у фігурі

Вхідні дані:

- r1: радіус внутрішнього кола (тип double), діапазон: $[0, +\infty)$.
- r2: радіус зовнішнього кола (тип double), діапазон: $[0, +\infty)$.
- a: відстань до центрів внутрішніх кіл (тип double), діапазон: $[0, +\infty)$.
- x: координата точки по осі X (тип double).
- y: координата точки по осі Y (тип double).

Вихідні дані:

- message: рядок (тип string), повідомлення про те, чи потрапляє точка в червоне або жовте коло.

Алгоритм вирішення:

1. Запитати у користувача значення r1, r2 і a.
2. Запросити координати точки x та y.
3. Обчислити відстані від точки до центрів червоних кіл і до зовнішнього жовтого кола.
4. Визначити, чи потрапляє точка в червоні або жовте коло.
5. Вивести відповідне повідомлення.

Екран роботи програми показаний в додатку Б, рис. Б.2(стор.8).

Завдання 3. Обчислення площі та периметру

Вхідні дані:

- radius: радіус кола (тип double), діапазон: $(0, +\infty)$.

Вихідні дані:

- area: площа кола (тип double).
- perimeter: периметр кола (тип double).

Алгоритм вирішення:

1. Запитати у користувача радіус кола radius.
2. Перевірити, чи радіус є додатним числом.
3. Обчислити площу кола за формулою: $\text{area} = \pi \times \text{radius}^2$
4. Обчислити периметр кола за формулою: $\text{perimeter} = 2 \times \pi \times \text{radius}$

5. Вивести значення площі та периметру.

Екран роботи програми показаний в додатку Б, рис. Б.3(стор.8).

Завдання 4. Меню для вибору задачі

Вирішення задачі: Розділ 1, номер 4

Вхідні дані:

- choice: вибір користувача (тип int), можливі значення: {0, 1, 2, 3}.

Вихідні дані:

- Відповідні текстові повідомлення про вибране завдання або вихід з програми.

Алгоритм вирішення:

1. Вивести меню з варіантами завдань.
2. Запросити у користувача вибір.
3. В залежності від вибору викликати відповідну функцію:
4. Завдання 1 — опис числа.
5. Завдання 2 — перевірка точки у фігурі.
6. Завдання 3 — обчислення площі та периметру.
7. Завдання 0 — вихід з програми.
8. Повторювати до тих пір, поки користувач не вибере вихід.

Екран роботи програми показаний в додатку Б, рис. Б.0(стор.7).

Лістинг коду вирішення задач наведено в додатку А (стор. 5).

Екран роботи програми показаний в додатку Б, рис. Б.0(стор.7).

ВИСНОВКИ

У ході виконання лабораторної роботи була розроблена програма на C++, яка складається з кількох задач. Загалом, робота поглибила мої знання з програмування, покращила навички алгоритмічного мислення та навчила ефективно взаємодіяти з користувачем. Було цікаво і водночас важкувато створювати просте меню, але я навчився це робити.

ДОДАТОК А

Лістинг коду програми

```

#include <iostream>
#include <cmath>

using namespace std;

// Завдання 1. Опис числа
void task_1() {
    int n;
    cout << "Введіть ціле число в діапазоні 1-999: ";
    cin >> n;

    if (n < 1 || n > 999) {
        cout << "Число поза діапазоном!" << endl;
        return;
    } else{

        // Визначаємо парність
        string parity = (n % 2 == 0) ? "парне" : "непарне";

        // Визначаємо кількість цифр
        string digits;
        if (n < 10) {
            digits = "однозначне";
        } else if (n < 100) {
            digits = "двозначне";
        } else {
            digits = "тризначне";
        }

        // Виводимо результат
        cout << parity << " " << digits << " число" << endl;
    }
}

// Завдання 2. Перевірка точки в області (коло)
void task_2() {
    double r1, r2, a;

    // Введення параметрів фігур
    cout << "Введіть радіус внутрішнього кола (r1): ";
    cin >> r1;
    cout << "Введіть радіус зовнішнього кола (r2): ";
    cin >> r2;
    cout << "Введіть відстань від центру зовнішнього кола до центру внутрішніх кіл (a): ";
    cin >> a;

    double x, y;
    cout << "Введіть координати точки (x, y): ";
    cin >> x >> y;

    // Центри червоних кіл
    double x1 = a * cos(M_PI / 4);
    double y1 = a * sin(M_PI / 4);
    double x2 = -a * cos(M_PI / 4);
    double y2 = -a * sin(M_PI / 4);

    // Перевірка потрапляння в перше червоне коло
    double distance1 = sqrt(pow(x - x1, 2) + pow(y - y1, 2));
    bool inFirstRedCircle = (distance1 <= r1);

    // Перевірка потрапляння в друге червоне коло
    double distance2 = sqrt(pow(x - x2, 2) + pow(y - y2, 2));
    bool inSecondRedCircle = (distance2 <= r1);
}

```

```

// Перевірка потрапляння в жовте коло
double distanceOuter = sqrt(pow(x, 2) + pow(y, 2));
bool inYellowCircle = (distanceOuter <= r2);

// Виведення результату
if (inFirstRedCircle || inSecondRedCircle) {
    cout << "Точка потрапляє в червоне коло" << endl;
} else if (inYellowCircle) {
    cout << "Точка потрапляє в жовте коло" << endl;
} else {
    cout << "Точка не потрапляє у фігуру" << endl;
}
}

// Завдання 3. Обчислення площі і периметру кола
void task_3() {
    double radius;
    cout << "Введіть радіус кола: ";
    cin >> radius;

    if (radius <= 0) {
        cout << "Радіус має бути більше нуля!" << endl;
        return;
    }

    // Обчислюємо площу та периметр
    double area = M_PI * radius * radius;
    double perimeter = 2 * M_PI * radius;

    // Виведення результатів
    cout << "Площа кола: " << area << endl;
    cout << "Периметр кола: " << perimeter << endl;
}

// Завдання 4. Меню
void menu() {
    int choice;

    do {
        // Виводимо меню
        cout << "\nОберіть завдання:\n";
        cout << "1. Опис числа\n";
        cout << "2. Перевірка точки у фігурі\n";
        cout << "3. Обчислення площі та периметру\n";
        cout << "0. Вихід\n";
        cout << "Введіть номер завдання: ";
        cin >> choice;

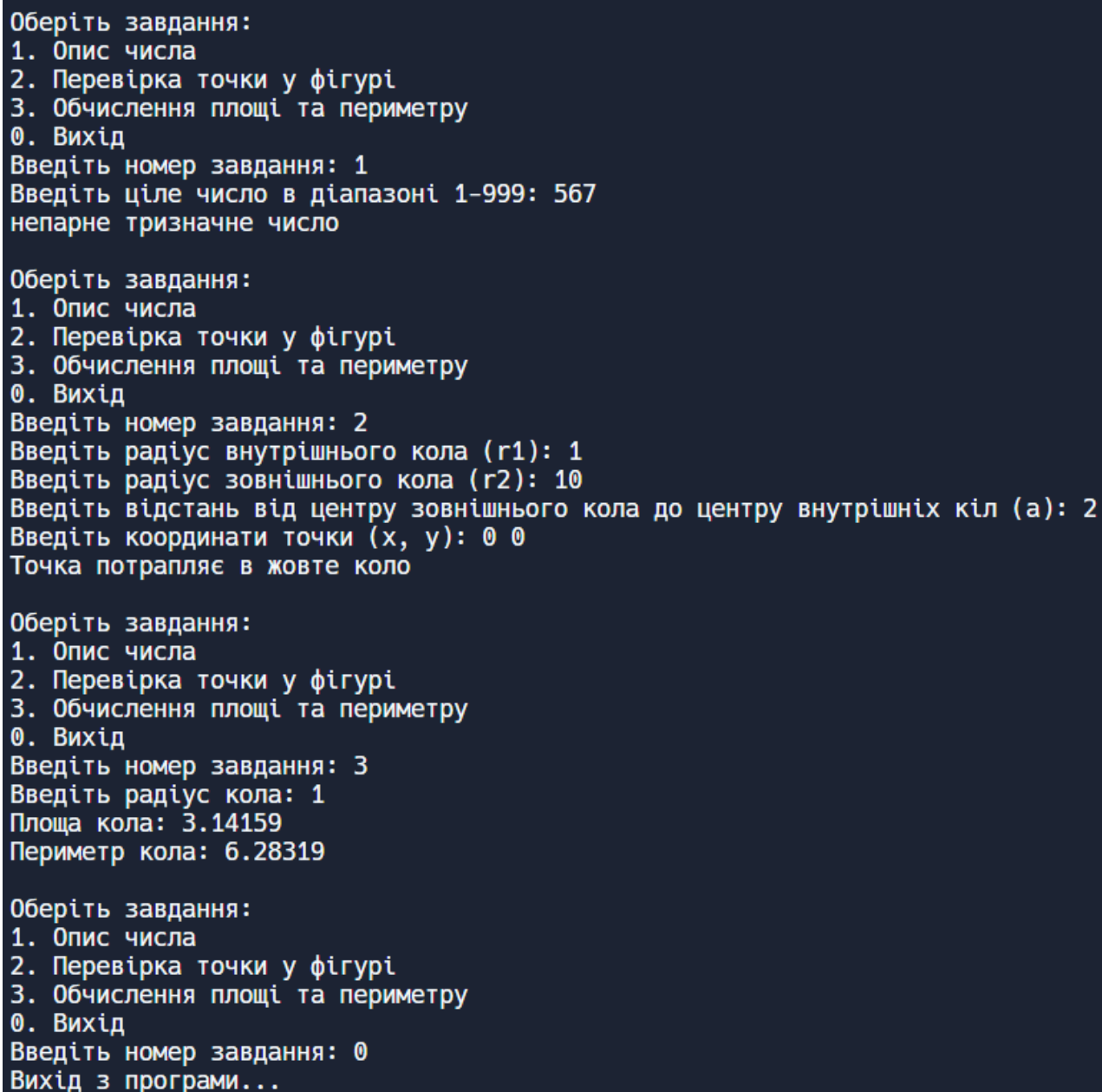
        // Викликаємо функцію залежно від вибору
        switch (choice) {
            case 1:
                task_1();
                break;
            case 2:
                task_2(); // Параметри вводимо тут
                break;
            case 3:
                task_3();
                break;
            case 0:
                cout << "Вихід з програми..." << endl;
                break;
            default:
                cout << "Невірний вибір. Спробуйте ще раз." << endl;
        }
    } while (choice != 0); // повторюємо, поки користувач не вибере вихід
}

```

```
int main() {  
    menu(); // Запуск меню  
    return 0;  
}
```

ДОДАТОК Б

Скрін-шоти вікна виконання програми



Оберіть завдання:
1. Опис числа
2. Перевірка точки у фігурі
3. Обчислення площі та периметру
0. Вихід
Введіть номер завдання: 1
Введіть ціле число в діапазоні 1-999: 567
непарне тризначне число

Оберіть завдання:
1. Опис числа
2. Перевірка точки у фігурі
3. Обчислення площі та периметру
0. Вихід
Введіть номер завдання: 2
Введіть радіус внутрішнього кола (r1): 1
Введіть радіус зовнішнього кола (r2): 10
Введіть відстань від центру зовнішнього кола до центру внутрішніх кіл (a): 2
Введіть координати точки (x, y): 0 0
Точка потрапляє в жовте коло

Оберіть завдання:
1. Опис числа
2. Перевірка точки у фігурі
3. Обчислення площі та периметру
0. Вихід
Введіть номер завдання: 3
Введіть радіус кола: 1
Площа кола: 3.14159
Периметр кола: 6.28319

Оберіть завдання:
1. Опис числа
2. Перевірка точки у фігурі
3. Обчислення площі та периметру
0. Вихід
Введіть номер завдання: 0
Вихід з програми...

Рисунок Б.0 – Екран виконання всієї програми

```
Оберіть завдання:  
1. Опис числа  
2. Перевірка точки у фігурі  
3. Обчислення площі та периметру  
0. Вихід  
Введіть номер завдання: 1  
Введіть ціле число в діапазоні 1-999: 1000  
Число поза діапазоном!
```

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

```
Оберіть завдання:  
1. Опис числа  
2. Перевірка точки у фігурі  
3. Обчислення площі та периметру  
0. Вихід  
Введіть номер завдання: 2  
Введіть радіус внутрішнього кола (r1): 1  
Введіть радіус зовнішнього кола (r2): 13  
Введіть відстань від центру зовнішнього кола до центру внутрішніх кіл (a): 0  
Введіть координати точки (x, y): 0 0  
Точка потрапляє в червоне коло
```

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

```
Оберіть завдання:  
1. Опис числа  
2. Перевірка точки у фігурі  
3. Обчислення площі та периметру  
0. Вихід  
Введіть номер завдання: 3  
Введіть радіус кола: 10  
Площа кола: 314.159  
Периметр кола: 62.8319
```

Рисунок Б.3 – Екран виконання програми для вирішення завдання 3