

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.301.173.310.1 ЛР

Виконав студент гр. _____ 310 _____

25.11.24 Андрій КОБИЛЯНСЬКИЙ
(підпис, дата) (П.І.Б.)

Перевірив

_____ к.т.н., доц. Олена ГАВРИЛЕНКО
(підпис, дата) (П.І.Б.)

2024

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.

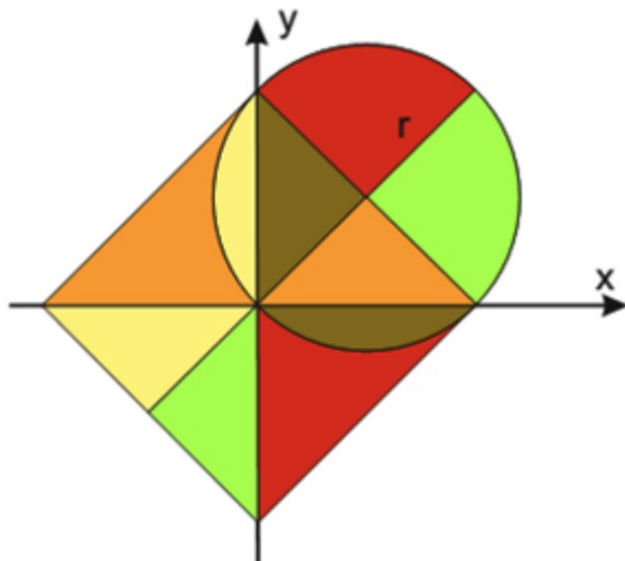
If21. Дано цілочисельні координати точки на площині. Якщо точка збігається з початком координат, то вивести 0. Якщо точка не збігається з початком координат, але лежить на осі OX або OY, то вивести відповідно 1 або 2. Якщо точка не лежить на координатних осях, то вивести 3.

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

Geom28.табл2.варіант 35 варіант, жовтий колір

r - радіус кола

-  **32 вар.**
-  **33 вар.**
-  **34 вар.**
-  **35 вар.**
-  **36 вар.**



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

Лістинг коду вирішення усіх задач наведено в дод. А (стор. 10-13).

Завдання 1.

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

If 21:

X – будь-яке ціле число, цілий тип, $A > 0$

Y – будь-яке ціле число, цілий тип, $B > 0$

Вихідні дані (ім'я, опис, тип):

Якщо точка збігається з початком координат, то вивести 0. Якщо точка не збігається з початком координат, але лежить на осі OX або OY, то вивести відповідно 1 або 2. Якщо точка не лежить на координатних осях, то вивести 3.

Алгоритм вирішення:

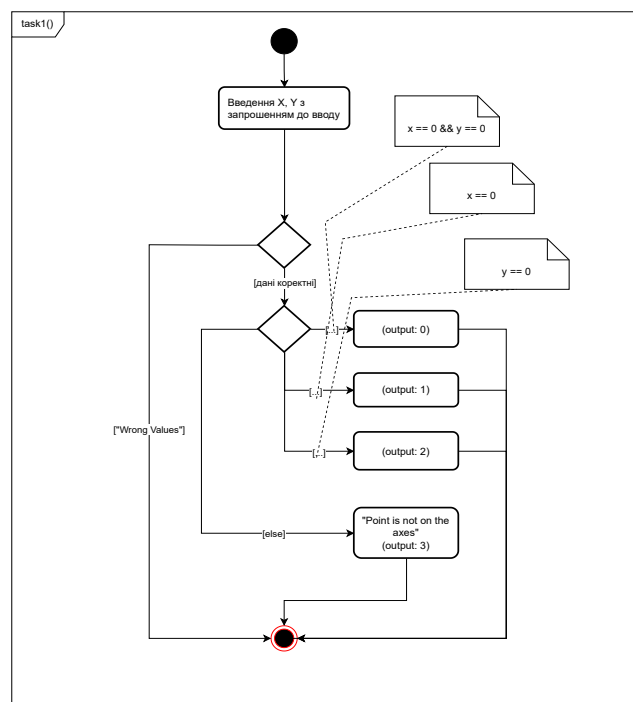


Рисунок 1 –
Діаграма
активності для
алгоритму
вирішення задачі
If 21:

Рисунок 2 – Екран роботи програми завдання If28:

```
_____Task 1 (if 21)_____
Enter X coordinate: 0
Enter Y coordinate: 0
Point is at the origin. Output: 0

Enter X coordinate: 0
Enter Y coordinate: -3
Point is on the Y-axis. Output: 1

Enter X coordinate: 32
Enter Y coordinate: 0
Point is on the X-axis. Output: 2

Enter X coordinate: 342
Enter Y coordinate: 5323
Point is not on the axes. Output: 3
```

Завдання 2.

Вирішення задачі №35

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

Geom 35: r – радіус кола, дійсний тип із подвійною точністю

x і y – координати точки, дійсний тип із подвійною точністю

Вихідні дані (ім'я, опис, тип):

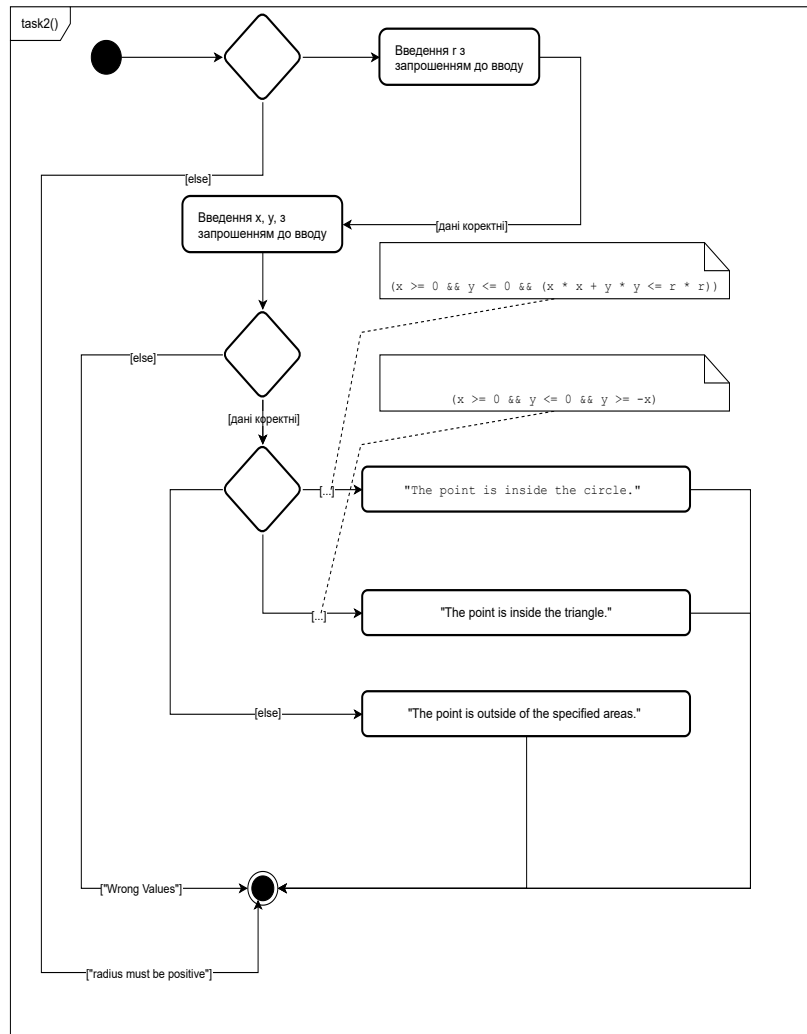
Якщо точка знаходиться в указаній площі ($x > 0$, $y < 0$ і $x^2 + y^2 \leq r^2$),

Виведення повідомлення о знаходженні.

Якщо ні, виведення повідомлення, що точка не знаходиться у фігурі.

Алгоритм вирішення:

Рисунок 4
— Діаграма активності для алгоритму вирішення задачі Geom 35:



Екран роботи програми показаний на рис. 5

```
_____Task 2 (35)_____
Enter the radius r: 65
Enter point coordinates x and y: 23
-10
The point is inside the circle.
```

Завдання 3.

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

r – радіус кола, дійсний тип із подвійною точністю

Вихідні дані (ім'я, опис, тип): Обчислення площі та периметра фігури

Area – площа фігури, дійсний тип із подвійною точністю

Perimeter - перимітер фігури, дійсний тип із подвійною точністю

Алгоритм вирішення:

- 1) Виведення запрошення до вводу
- 2) Введення змінної r
- 3) Обчислення площі чверті круга
- 4) Обчислення периметра (довжини дуги + 2 радіуси)
- 5) Виведення результату площі та периметра жовтої фігури

Рисунок 6 – Діаграма активності для алгоритму вирішення задачі Geom 35:

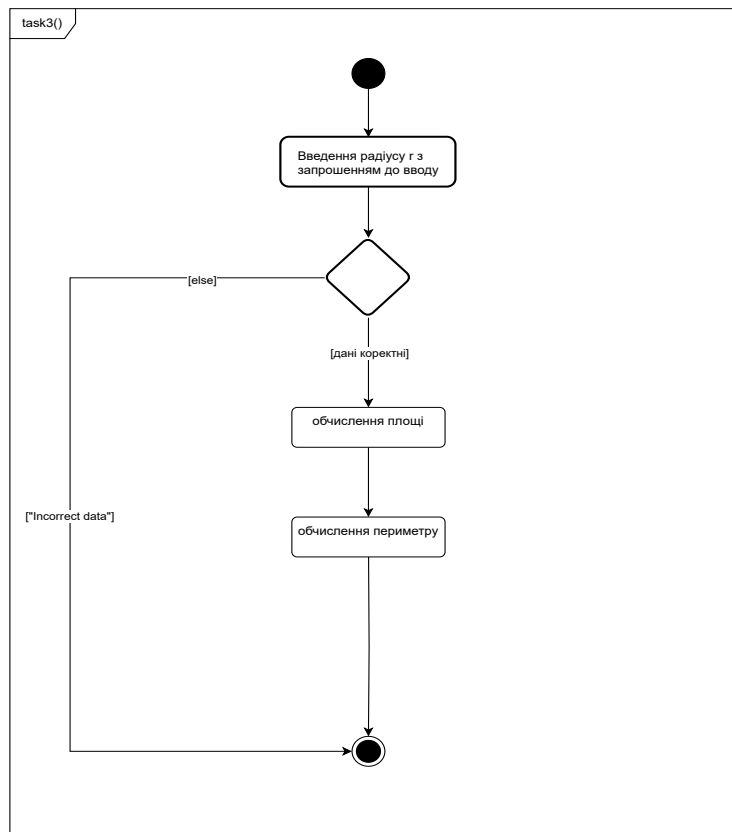


Рисунок
7 – Екран
роботи
програми

завдання Geom28, обчислення площі та
периметра:

```

Enter task number: 3

_____Task 3 (35)_____

Enter the radius r:
48
Area of the figure: 2961.56

Perimeter of the figure: 171.398
  
```

Завдання 4.

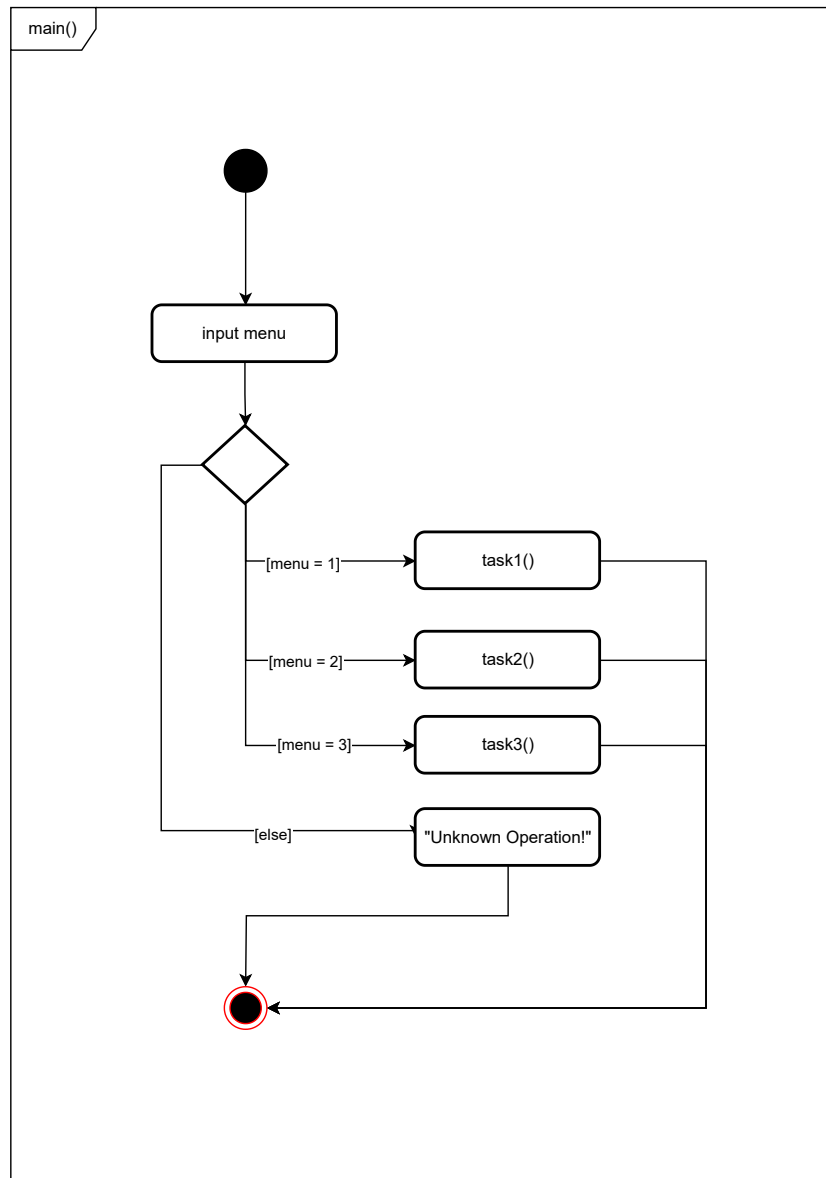
Вирішення задачі

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

menu – ціле число від 1 до 3 дійсний тип

Вихідні дані (ім'я, опис, тип): Виведення завдання

Рисунок 6 – Діаграма активності для алгоритму вирішення задачі:



ВИСНОВКИ

Було вивчено програму з трьома завданнями, кожне з яких демонструє певний

аспект роботи з умовами, геометрією та обчисленнями. Користувач може обрати одну з трьох задач через меню в головній функції. Завдання 1 (task_if10). Програма порівнює два числа A і B. Якщо вони не рівні, обчислюється їх сума, і обидві змінні отримують це значення. У випадку рівності значень обидві змінні скидаються в нуль. Завдання демонструє використання умовних операторів (if-else) для зміни значень змінних.

Завдання 2 (task_geom28) Програма визначає, чи входить точка з координатами (x, y) в задану червону фігуру. Фігура складається з двох частин:

частини круга і частини прямокутника. Використовується перевірка через математичні нерівності та формулу для кола. У разі виконання умов виводиться повідомлення, чи належить точка заданій області. Завдання 3 (task_calculate28) Виконується обчислення площі та периметра червоної фігури з використанням формул. Розрахунки базуються на відомих геометричних формулах для круга, трикутника та комбінації їх частин.

Значення площі та периметра виводяться в консоль.

ДОДАТОК А

Лістинг коду програми.

```
#include <iostream>
#include <cmath>

using namespace std;
void task2();

void task3();

int main()
{
```

```

int menu;
cout << "\nEnter task number: ";
cin >> menu;
switch (menu)
{
    case 1 : task1(); break; //1st Task

    case 2 : task2(); break; //2st Task

    case 3 : task3(); break; //3st Task

    default : cout << "\nUnknown operation!";

}
return 0;
}

void task1()

{
    /*
    Task 1 if21:
    Дано цілочисельні координати точки на площині. Якщо точка збігається з
    початком координат, то вивести 0. Якщо точка не збігається з початком
    координат,
    але лежить на осі OX або OY, то вивести відповідно 1 або 2. Якщо точка не
    лежить
    на координатних осях, то вивести 3.
    */

    cout << "\n_____Task 1 (if 21)_____" << endl << endl;

    // Declaration of coordinates
    int x, y;

    // Input for X and Y coordinates
    cout << "Enter X coordinate: ";
    cin >> x;
    cout << "Enter Y coordinate: ";
    cin >> y;

    if (cin)
    {
        // Check the position of the point
        if (x == 0 && y == 0)
        {
            // If the point is at the origin
            cout << "Point is at the origin. Output: 0" << endl << endl;

```

```

    } else if (x == 0)
    {
        // If the point is on the Y-axis
        cout << "Point is on the Y-axis. Output: 1" << endl << endl;
    } else if (y == 0)
    {
        // If the point is on the X-axis
        cout << "Point is on the X-axis. Output: 2" << endl << endl;
    }
    else
    {
        // If the point is not on any axis
        cout << "Point is not on the axes. Output: 3" << endl << endl;
    }
    }
    else
    {
        cout << "\n\nWrong values! " << endl << endl;
    }
}

void task2()
{
    //Task 2-3 (35) Таблиця 2 Геометричні фігури
    cout << "\n_____Task 2 (35)_____" << endl << endl;
    // Declaration:
    float x, y, r;
    cout << "Enter the radius r: ";
    cin >> r;

    // Перевірка на валідність введених даних
    if (r <= 0)
    {
        cout << "Radius must be positive!" << endl;
    }

    // Введення координат точки
    cout << "Enter point coordinates x and y: ";
    cin >> x >> y;

    // Умова для перевірки частини кола (четвертина кола у 4-й чверті)
    if (x >= 0 && y <= 0 && (x * x + y * y <= r * r)) {
        cout << "The point is inside the circle." << endl;
    }
    // Умова для трикутника
    else if (x >= 0 && y <= 0 && y >= -x) {
        cout << "The point is inside the triangle." << endl;
    }
    // Якщо точка не належить жодній із областей

```

```

    else {
        cout << "The point is outside of the specified areas." << endl;
    }
}

void task3()
{
    float r;

    cout << "Enter the radius r: " << endl;
    cin >> r;

    // Перевірка на валідність введених даних
    if (!cin || r <= 0)
        cout << "Incorrect data!" << endl << endl; //повідомлення про помилку
    else
    {
        // Обчислення площі
        float sectorArea = M_PI * r * r / 4;
        float triangleArea = r * r / 2;
        float totalArea = sectorArea + triangleArea;

        // Обчислення периметра
        float arcLength = M_PI * r / 2;
        float totalPerimeter = arcLength + 2 * r;

        // Виведення результатів
        cout << "Area of the figure: " << totalArea << endl << endl;
        cout << "Perimeter of the figure: " << totalPerimeter << endl << endl;
    }
}

```

