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 С ++»

Виконав студент гр. \_\_\_\_\_\_319\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*Микита Тiтов\_\_\_\_\_\_\_*

(підпис, дата) (П.І.Б.)

Перевірив

\_\_\_\_\_\_\_\_\_ к.т.н., доц. Олена  ГАВРИЛЕНКО

(підпис, дата) (П.І.Б.)

2023

# 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 C++ в середовищі Visual Studio. Також опанувати та відпрацювати навички структурування програми з функціями.

# ПОСТАНОВКА ЗАДАЧІ

Завдання 1. Вирішити завдання 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. Обчислити площу і периметр плоскої фігури.

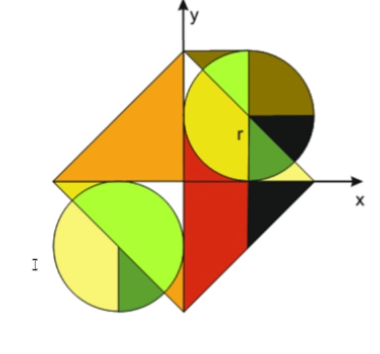
Завдання 4. Для вибору користувачем одного з трьох зазначених вище завдань розробити алгоритм організації меню в командному вікні з використанням інструкції вибору.

# ВИКОНАННЯ РОБОТИ

Вирішення задачі If 28; 16 варiант;

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

Дано номер року (додатне ціле число). Визначити кількість днів в цьому році, враховуючи, що звичайний рік нараховує 365 днів, а високосний - 366 днів. Високосним вважається рік, ділиться на 4, за винятком тих років, які діляться на 100 і не діляться на 400 (наприклад, роки 300 1300 і 1900 не є високосними, а 1200 і 2000 - є);



Вихідні дані (ім’я, опис, тип):

Змінні:

PI:

Тип: const double.

Константа, що представляє значення числа π з високою точністю (3.14159...).

Використовується для розрахунків площі, периметра та довжини дуги.

year:

Тип: int.

Зберігає номер року, введеного користувачем.

Використовується для обчислення кількості днів у році.

days:

Тип: int.

Результат функції daysInYear, що повертає кількість днів у році (365 або 366).

x, y:

Тип: double.

Координати точки, введені користувачем.

Використовуються для перевірки, чи належить точка заданій фігурі.

r:

Тип: double.

Радіус кола, введений користувачем.

Використовується для обчислення площі, периметра фігури та перевірки належності точки.

sectorArea:

Тип: double.

Площа сектора кола (чверті кола) з радіусом r.

triangleArea:

Тип: double.

Площа прямокутного трикутника зі сторонами, рівними r.

arcLength:

Тип: double.

Довжина дуги сектора (чверті кола) з радіусом r.

triangleSides:

Тип: double.

Загальна довжина двох рівних сторін прямокутного трикутника (обидві дорівнюють r).

area:

Тип: double.

Площа об'єднаної фігури (сектора + трикутника).

perimeter:

Тип: double.

Периметр об'єднаної фігури (дуга + сторони трикутника).

Типи даних:

int:

Використовується для роботи з цілими числами, такими як рік та кількість днів.

double:

Використовується для роботи з дійсними числами (координати, радіус, площа, периметр, π).

const:

Використовується для визначення константного значення (наприклад, PI), яке не змінюється під час виконання програми.

Текстові сповіщення:

cout << "Введите номер года: ";

Запитує у користувача номер року для обчислення кількості днів.

cout << "Количество дней в " << year << " году: " << days << endl;

Виводить кількість днів у введеному році, залежно від того, чи є він високосним.

cout << "Введите координаты точки (x, y): ";

Запитує у користувача координати точки.

cout << "Введите радиус r: ";

Запитує у користувача радіус фігури.

cout << "Точка принадлежит зеленому сектору (второй квадрант)." << endl;

Інформує користувача, що точка належить сектору (друга чверть координатної площини).

cout << "Точка принадлежит зеленому треугольнику (третий квадрант)." << endl;

Інформує користувача, що точка належить трикутнику (третя чверть координатної площини).

cout << "Точка не принадлежит ни одной из фигур." << endl;

Вказує, що точка не належить жодній із заданих фігур.

cout << "Площадь фигуры: " << area << endl;

Виводить обчислену площу об'єднаної фігури.

cout << "Периметр фигуры: " << perimeter << endl;

Виводить обчислений периметр об'єднаної фігури.

Робота функцій:

daysInYear(int year):

Перевіряє, чи є рік високосним, використовуючи правила:

Високосний, якщо ділиться на 4, але не ділиться на 100.

Або ділиться на 400.

Повертає кількість днів у році (366 або 365).

calculateArea(double r):

Обчислює площу об'єднаної фігури:

Площа сектора кола: πr24\frac{\pi r^2}{4}4πr2​.

Площа прямокутного трикутника: r22\frac{r^2}{2}2r2​.

calculatePerimeter(double r):

Обчислює периметр об'єднаної фігури:

Довжина дуги сектора: πr2\frac{\pi r}{2}2πr​.

Дві рівні сторони трикутника: 2r2r2r.

Алгоритм вирішення

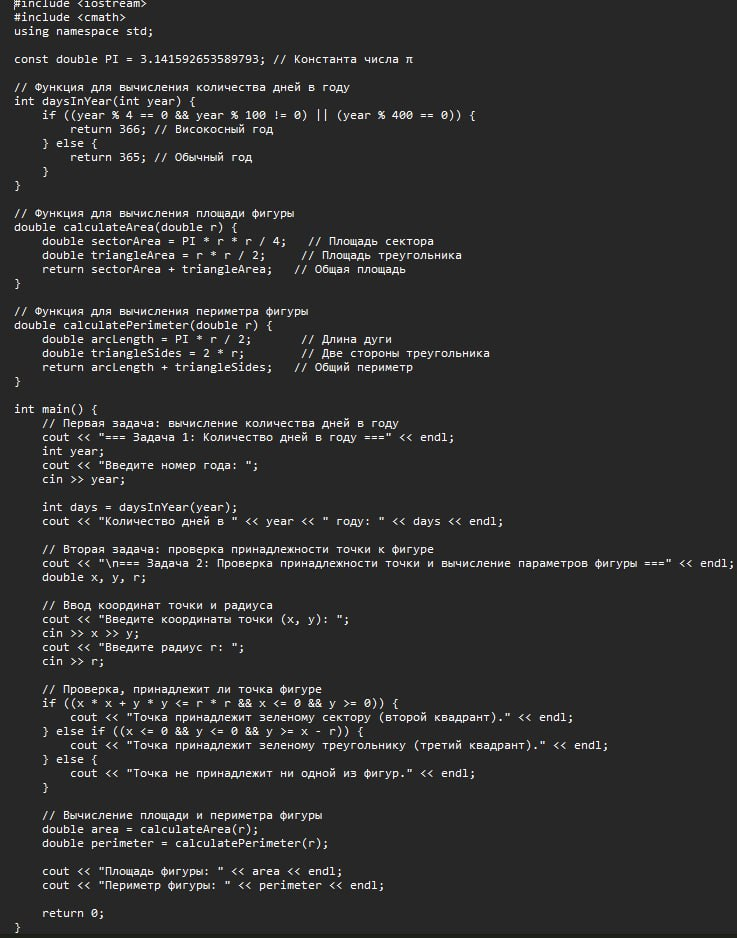


Рисунок 1 – Алгоритм вирiшення

+--------------------+

| Початок |

+--------------------+

|

v

+--------------------+

| Введення року |

| (year = 2025) |

+--------------------+

|

v

+--------------------+

| Виклик функції |

| days\_in\_year(year)|

+--------------------+

|

v

+--------------------+

| Виклик функції |

| is\_leap\_year(year)|

+--------------------+

|

v

+--------------------+

| Перевірка умов для |

| високосного року |

+--------------------+

|

v

+--------------------+

| Рік високосний? |

+--------------------+

| | |

| v |

| +--------------|-----------------+

| | v |

| | +------------------------+ |

| | | Повертає True | |

| | +------------------------+ |

| | | |

| | v |

| | +------------------------+ |

| | | Повертає 366 днів | |

| | +------------------------+ |

| | | |

| | v |

| | +------------------------+ |

| | | Виведення результату | |

| | +------------------------+ |

| | | |

| | v |

| | +------------------------+ |

| | | Кінець | |

| | +------------------------+ |

| | |

| +-----------------------------+

| |

| v

| +------------------------+

| | Повертає False |

| +------------------------+

| |

| v

| +------------------------+

| | Повертає 365 днів |

| +------------------------+

| |

| v

| +------------------------+

| | Виведення результату |

| +------------------------+

| |

| v

| +------------------------+

| | Кінець |

| +------------------------+

|

+-----------------------------+

Дiаграмма 1. Дiаграмма активностi

Лістинг коду вирішення задачі:

#include <iostream>

#include <cmath>

using namespace std;

const double PI = 3.141592653589793; // Константа числа π

// Функция для вычисления количества дней в году

int daysInYear(int year) {

if ((year % 4 == 0 && year % 100 != 0) || (year % 400 == 0)) {

return 366; // Високосный год

} else {

return 365; // Обычный год

}

}

// Функция для вычисления площади фигуры

double calculateArea(double r) {

double sectorArea = PI \* r \* r / 4; // Площадь сектора

double triangleArea = r \* r / 2; // Площадь треугольника

return sectorArea + triangleArea; // Общая площадь

}

// Функция для вычисления периметра фигуры

double calculatePerimeter(double r) {

double arcLength = PI \* r / 2; // Длина дуги

double triangleSides = 2 \* r; // Две стороны треугольника

return arcLength + triangleSides; // Общий периметр

}

int main() {

// Первая задача: вычисление количества дней в году

cout << "=== Задача 1: Количество дней в году ===" << endl;

int year;

cout << "Введите номер года: ";

cin >> year;

int days = daysInYear(year);

cout << "Количество дней в " << year << " году: " << days << endl;

// Вторая задача: проверка принадлежности точки к фигуре

cout << "\n=== Задача 2: Проверка принадлежности точки и вычисление параметров фигуры ===" << endl;

double x, y, r;

// Ввод координат точки и радиуса

cout << "Введите координаты точки (x, y): ";

cin >> x >> y;

cout << "Введите радиус r: ";

cin >> r;

// Проверка, принадлежит ли точка фигуре

if ((x \* x + y \* y <= r \* r && x <= 0 && y >= 0)) {

cout << "Точка принадлежит зеленому сектору (второй квадрант)." << endl;

} else if ((x <= 0 && y <= 0 && y >= x - r)) {

cout << "Точка принадлежит зеленому треугольнику (третий квадрант)." << endl;

} else {

cout << "Точка не принадлежит ни одной из фигур." << endl;

}

// Вычисление площади и периметра фигуры

double area = calculateArea(r);

double perimeter = calculatePerimeter(r);

cout << "Площадь фигуры: " << area << endl;

cout << "Периметр фигуры: " << perimeter << endl;

return 0;

}}

Екран роботи програми показаний на рис.

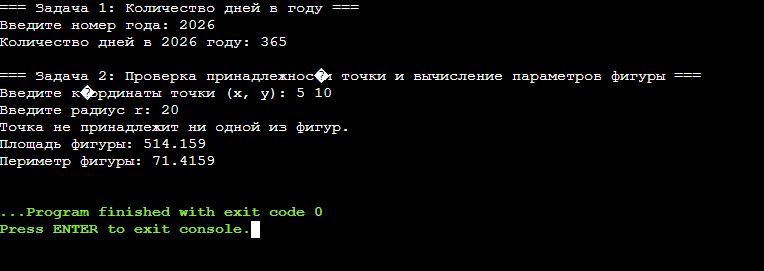


Рисунок 2. Екран роботи програми

# ВИСНОВКИ

Пiд час роботи вивчили теоретичний матеріал щодо синтаксису у мові С ++ і подання у вигляді UML діаграм активності алгоритмів з розгалуженням та реалізували алгоритми з використанням інструкцій умовного переходу і вибору мовою C++ в середовищі Visual Studio. Також опанували та відпрацювати навички структурування програми з функціями.

ДОДАТОК А

Лістинг коду програми

#include <iostream>

#include <cmath>

using namespace std;

const double PI = 3.141592653589793; // Константа числа π

// Функция для вычисления количества дней в году

int daysInYear(int year) {

if ((year % 4 == 0 && year % 100 != 0) || (year % 400 == 0)) {

return 366; // Високосный год

} else {

return 365; // Обычный год

}

}

// Функция для вычисления площади фигуры

double calculateArea(double r) {

double sectorArea = PI \* r \* r / 4; // Площадь сектора

double triangleArea = r \* r / 2; // Площадь треугольника

return sectorArea + triangleArea; // Общая площадь

}

// Функция для вычисления периметра фигуры

double calculatePerimeter(double r) {

double arcLength = PI \* r / 2; // Длина дуги

double triangleSides = 2 \* r; // Две стороны треугольника

return arcLength + triangleSides; // Общий периметр

}

int main() {

// Первая задача: вычисление количества дней в году

cout << "=== Задача 1: Количество дней в году ===" << endl;

int year;

cout << "Введите номер года: ";

cin >> year;

int days = daysInYear(year);

cout << "Количество дней в " << year << " году: " << days << endl;

// Вторая задача: проверка принадлежности точки к фигуре

cout << "\n=== Задача 2: Проверка принадлежности точки и вычисление параметров фигуры ===" << endl;

double x, y, r;

// Ввод координат точки и радиуса

cout << "Введите координаты точки (x, y): ";

cin >> x >> y;

cout << "Введите радиус r: ";

cin >> r;

// Проверка, принадлежит ли точка фигуре

if ((x \* x + y \* y <= r \* r && x <= 0 && y >= 0)) {

cout << "Точка принадлежит зеленому сектору (второй квадрант)." << endl;

} else if ((x <= 0 && y <= 0 && y >= x - r)) {

cout << "Точка принадлежит зеленому треугольнику (третий квадрант)." << endl;

} else {

cout << "Точка не принадлежит ни одной из фигур." << endl;

}

// Вычисление площади и периметра фигуры

double area = calculateArea(r);

double perimeter = calculatePerimeter(r);

cout << "Площадь фигуры: " << area << endl;

cout << "Периметр фигуры: " << perimeter << endl;

return 0;

}

ДОДАТОК Б

Скрін-шоти вікна виконання програми

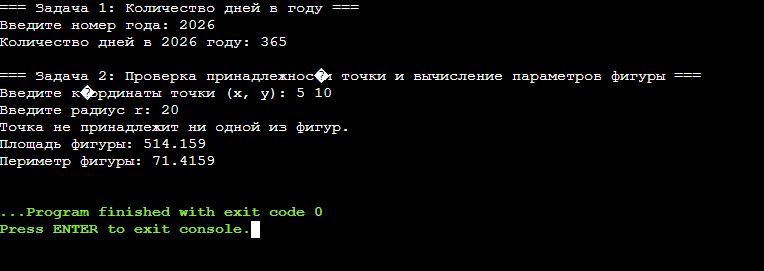


Рисунок Б.1 – Екран виконання програми для вирішення завдання