

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 ++»

ХАІ.301. Електрична інженерія. 319а. 14 ЛР

Виконав студент гр. 319а

29.10.2025
(підпис, дата)

Володимир Івахнін
(П.І.Б.)

Перевірів
29.10.2025
(підпис, дата)

асистент Євгеній Пявка
(П.І.Б.)

МЕТА РОБОТИ

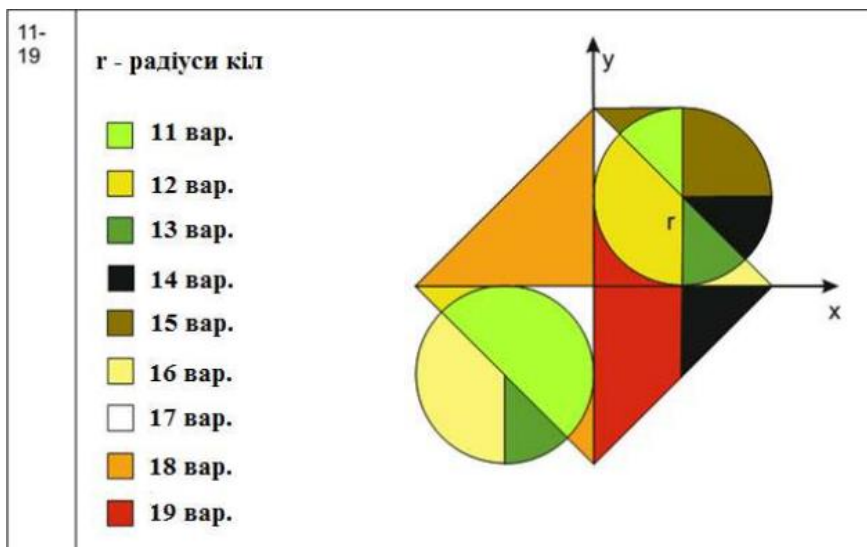
Вивчити теоретичний матеріал щодо синтаксису у мові C++ і подання у вигляді UML діаграм активності алгоритмів з розгалуженням та реалізувати алгоритми з використанням інструкцій умовного переходу і вибору мовою C++ в середовищі QtCreator. Також опанувати та відпрацювати навички структурування програми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

ПОСТАНОВКА ЗАДАЧІ

Вирішити дві задачі на алгоритми з розгалуженням.

Завдання 1. If28. 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).

Завдання 2. Figure 13. Дано координати точки на площині (x, y). Визначити, чи потрапляє точка в фігуру заданого кольору (або групу фігур) і вивести відповідне повідомлення.



ВИКОНАННЯ РОБОТИ

Завдання 1.

Вирішення задачі If 28.

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

Дано номер року (додатне ціле число).

Вихідні дані (ім'я, опис, тип):

Визначити кількість днів в цьому році,

враховуючи, що звичайний рік нараховує 365 днів, а високосний - 366 днів.

Лістинг коду вирішення задачі наведено в дод. А (стор. 7).

Екран роботи програми показаний на рис. Б.1

Завдання 2.

Вирішення задачі Figure 13.

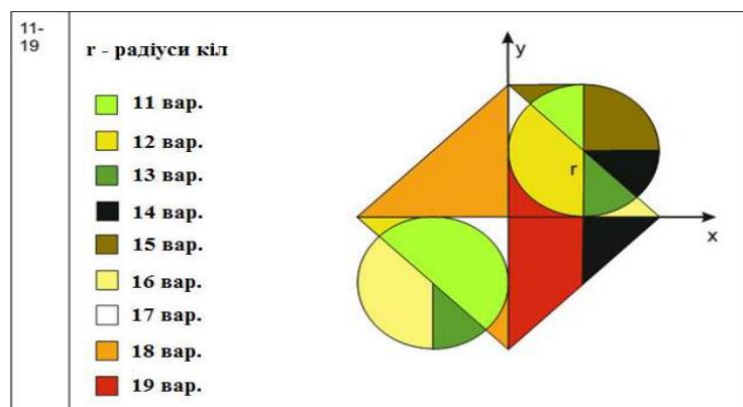
Вхідні дані (ім'я, опис, тип, обмеження):

Дано координати точки на площині (x, y).

Вихідні дані (ім'я, опис, тип):

Визначити, чи

потрапляє точка в фігуру заданого кольору (або групу фігур) і вивести відповідне повідомлення.



ВИСНОВКИ

Ми вивчили основи синтаксису мови C++, зокрема інструкції умовного переходу та вибору, а також принципи структурування програми за допомогою функцій. Також було розглянуто створення UML діаграм активності для алгоритмів з розгалуженням і реалізація цих алгоритмів у середовищі Qt Creator.

ДОДАТОК А

Лістинг коду програми

```
// Example program
#include <iostream>
using namespace std;

//If1.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).
void task_if128(); // 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_geom13();// завдання 2 декларація функції

int main() {
    int menu;
    cout << "Task number: ";
    cin >> menu;

    switch (menu)
    {
        case 1:task_if128();break; // Завдання 1
        case 2:task_geom13();break; // Завдання 2
        default: cout << "Wrong task! (Only 1,2)" << endl; //повідомлення про
помилку
    }
    system ("pause");
    return 0;
}

// Завдання 1: визначити кількість днів у році
void task_if128()
{
    int year;
    cout << "Введіть рік: ";
    cin >> year;

    if (cin.fail()) {
        cout << "Невірне введення!" << endl;
        return ;
    }
}
```

```

bool leap = (year % 4 == 0 && year % 100 != 0) || (year % 400 == 0);

if (leap)
    cout << "Цього року 366 днів (високосний рік)." << endl;
else
    cout << "Цей рік має 365 днів (звичайний рік)." << endl;
}

// Завдання 2: варіант 13 - перевірити належність точки області
void task_geom13() {
    double x, y, r;

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

    if (cin.fail() || r <= 0) {
        cerr << "Некоректні дані!" << endl;
        return;
    }
    // Ліве коло: центр (-r, 0)
    double circleLeft = pow(x + r, 2) + pow(y, 2);
    // Праве коло: центр (r, 0)
    double circleRight = pow(x - r, 2) + pow(y, 2);

    bool inLeftCircle = circleLeft <= r * r;
    bool inRightCircle = circleRight <= r * r;

    // Темно-зелена область — перетин двох кіл
    bool inDarkGreen = inLeftCircle && inRightCircle;

    if (inDarkGreen) {
        cout << "Точка належить області 13 (темно-зелена частина)." << endl;
    } else {
        cout << "Точка НЕ належить області 13." << endl;
    }
    return;
}

```

ДОДАТОК Б

Скрін-шоти вікна виконання програми

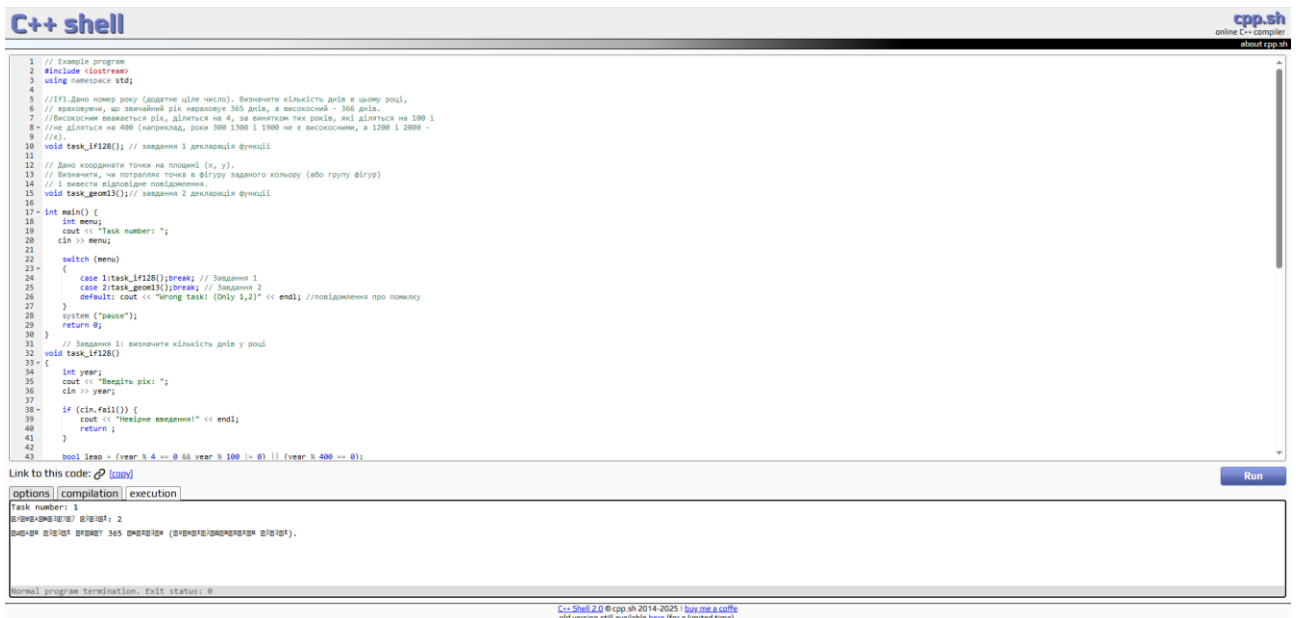


Рисунок Б.1 – Екран виконання програми для вирішення завдання

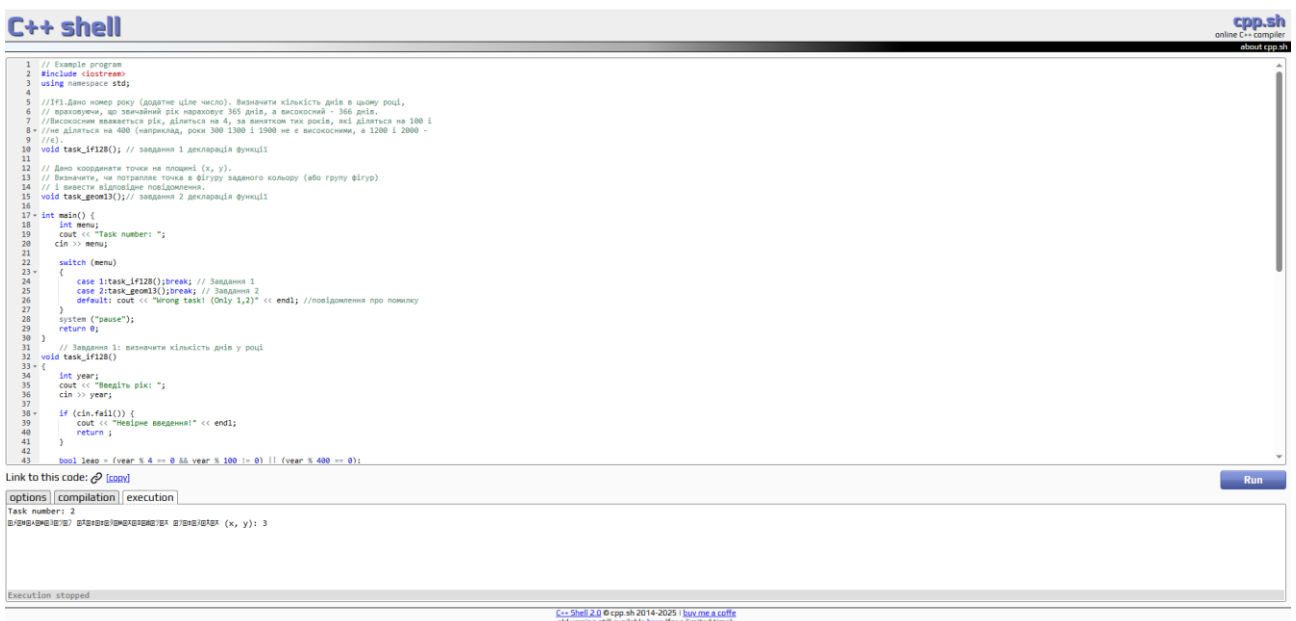
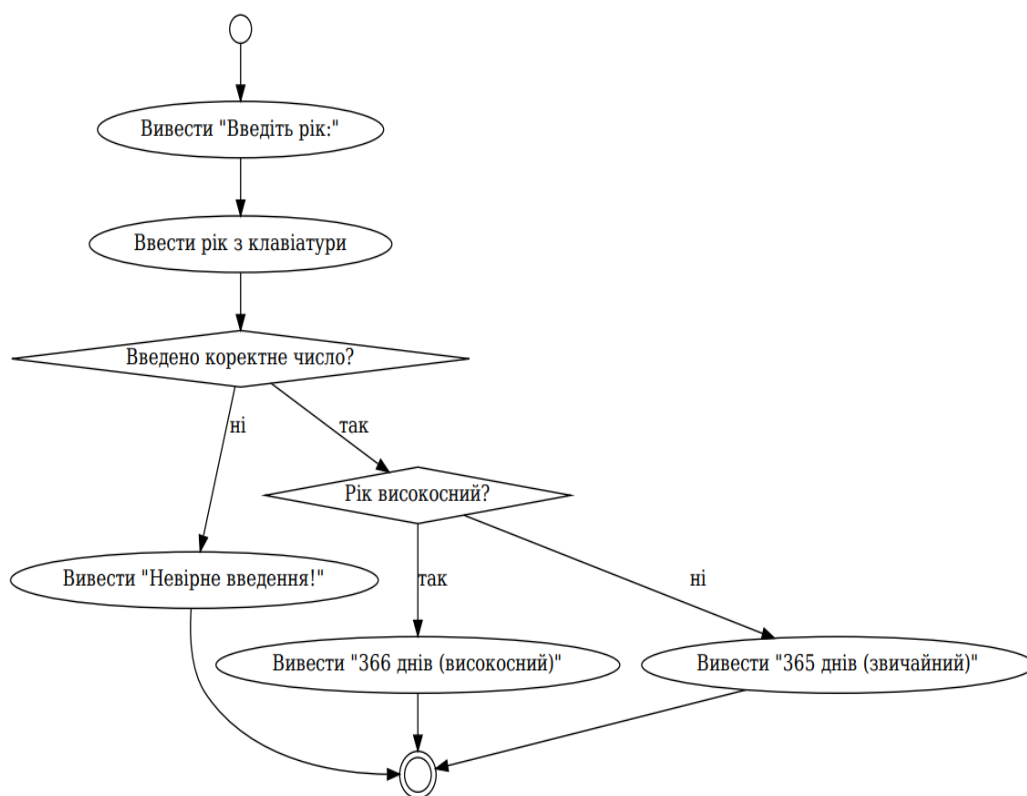
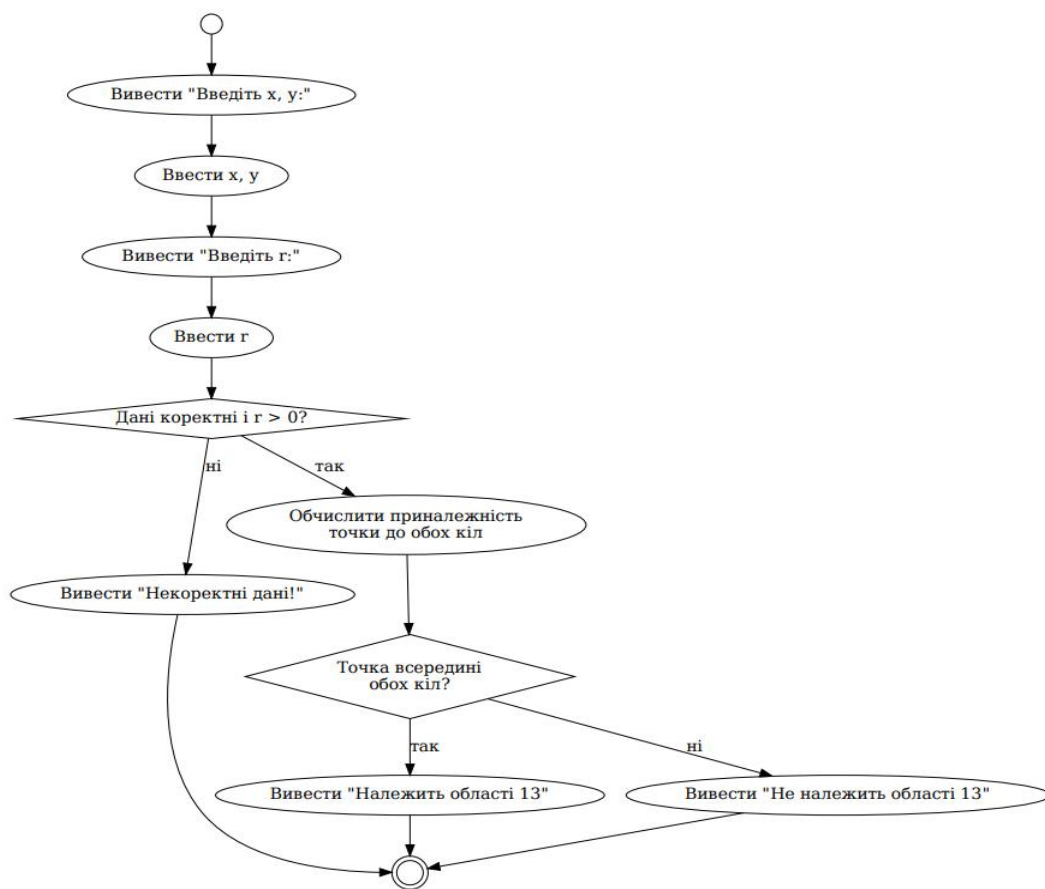


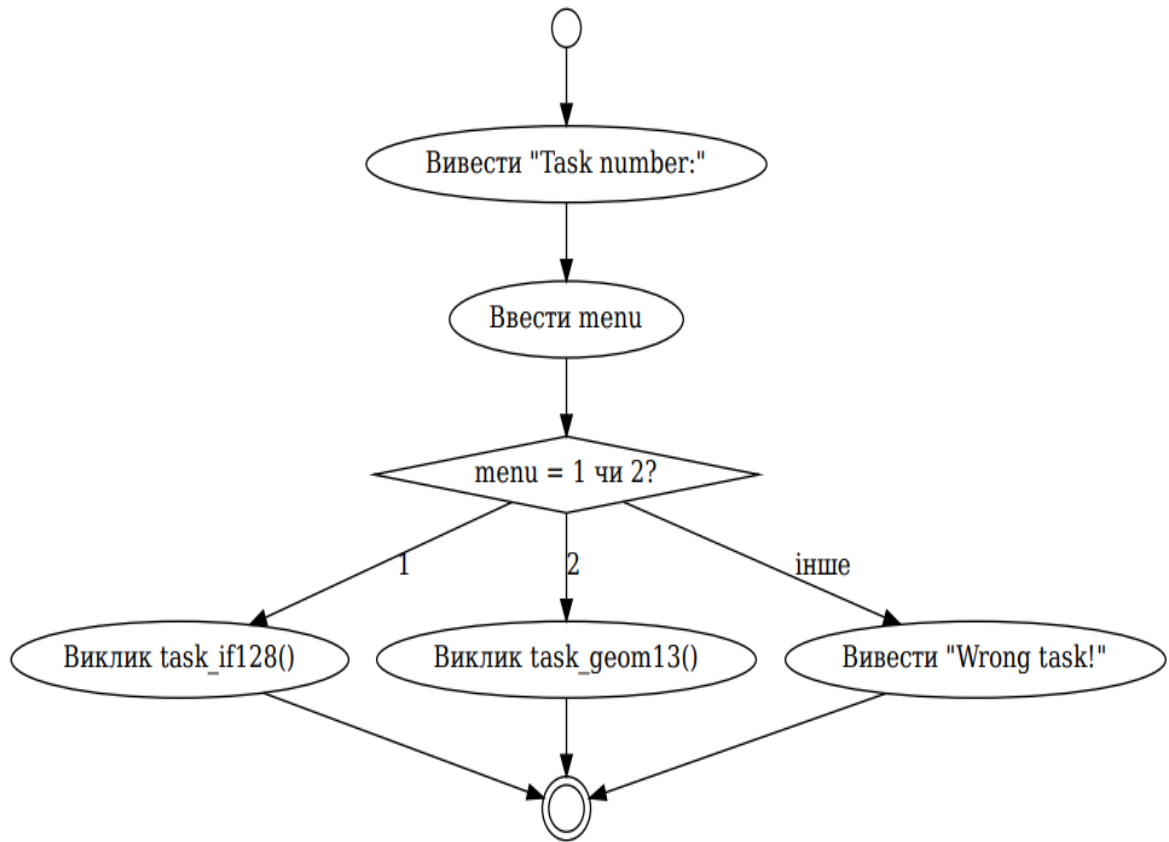
Рисунок Б.2 – Екран виконання програми для вирішення завдання
Figure 13.



Task_If28



Task_geom13



Main

ДОДАТОК В

Які основні концепції мови C++, які ти засвоїв під час виконання цього завдання?

Я навчився використовувати інструкції умовного переходу (if, else) та оператор вибору switch, а також структурувати код за допомогою функцій.

Як ти оцінюєш своє розуміння UML діаграм активності?

UML діаграми допомагають візуалізувати логіку алгоритмів, особливо з розгалуженнями, і полегшують планування процесів.

Як ти застосовував знання на практиці в Qt Creator?

Я створив консольну програму для перевірки парності числа, використовуючи умови та структуруючи код у функції.

Чи були труднощі під час роботи? Як ти їх подолав?

Було складно правильно відобразити алгоритм на UML, але я зрозумів, як це робити, переглянувши приклади.

Що ти хочеш покращити в майбутньому?

Хочу вдосконалити навички створення складніших алгоритмів та поглибити знання з Qt і графічних інтерфейсів.