

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. 175. 318. 02 ЛР

Виконав студент гр. _____ 318

_____ Бондаренко С. Б.
(підпис, дата) (П.І.Б.)

Перевірив

_____ к.т.н., доц. Гавриленко О. В.
(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.

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

Завдання 1.

Вирішення задачі If 28

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

year — рік, що перевіряється, int. Додатнє ціле число

Вихідні дані (ім'я, опис, тип):

isLeapYear — чи є рік високосним (true або false), bool.

message — повідомлення про тип року, string.

На рис. 1 показаний приклад діаграми для завдання If 28.

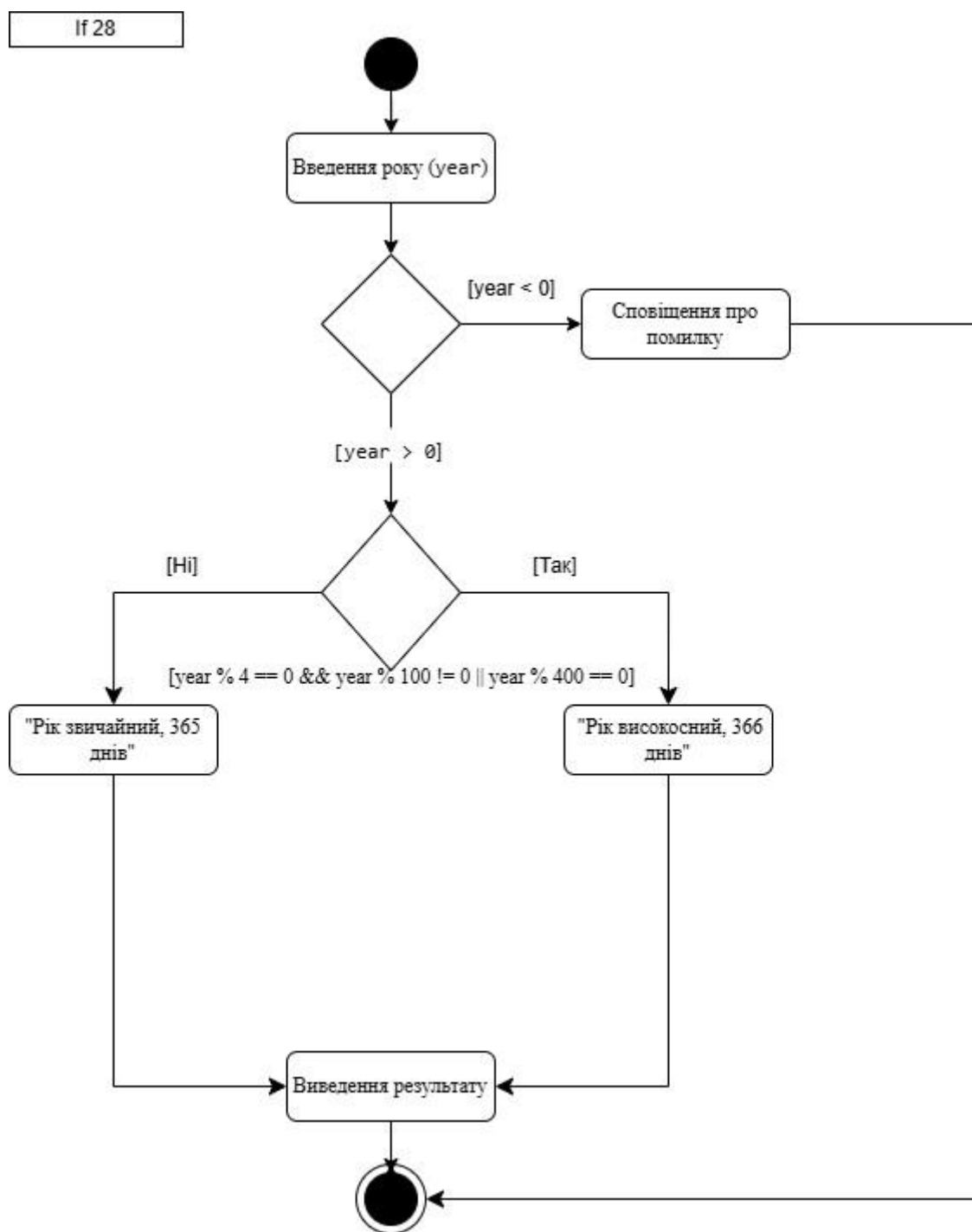


Рисунок 1 – Діаграма для завдання If 28

Лістинг коду вирішення задачі If 28 наведено в дод. А (стор. 7).

Екран роботи програми показаний на рис. Б.1.

Завдання 2, 3.

Вирішення задачі Geom 10.

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

x — координата x точки, double, довільне дійсне число.

y — координата y точки, double, довільне дійсне число.

a — сторона квадрата, double, додатне число більше нуля.

Вихідні дані (ім'я, опис, тип):

brownArea — площа коричневої області, обчислена як різниця між площею сектора та площі трикутника, double.

brownPerimeter — периметр коричневої області, обчислений як сума довжини дуги сектора та двох радіусів, double.

isPointInside — результат перевірки, чи належить точка коричневій області, bool.

На рис. 2 показаний приклад діаграми для завдання Geom 10.

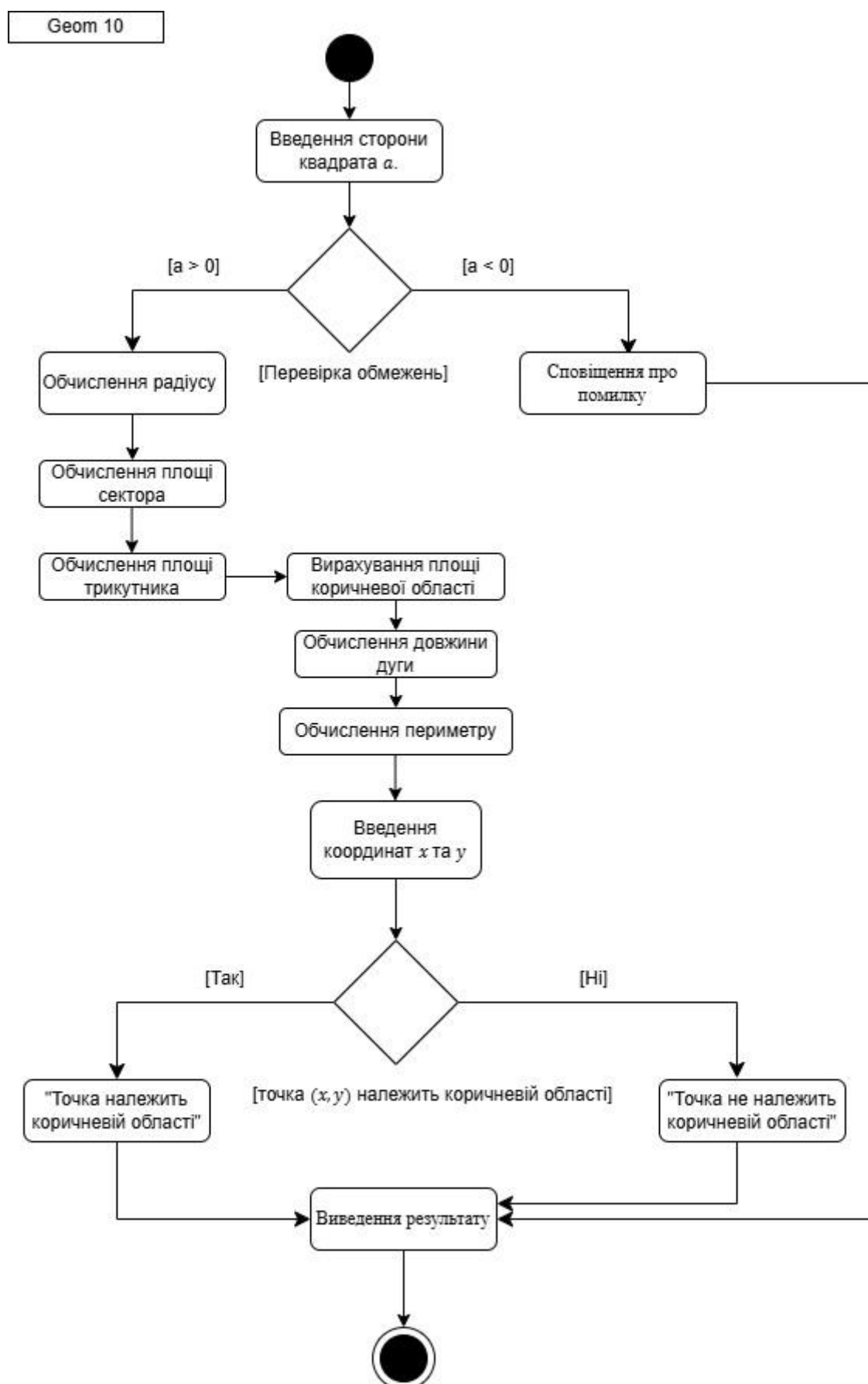


Рисунок 2 – Діаграма для завдання Geom 10

Лістинг коду вирішення задачі Geom 10 наведено в дод. А (стор. 7).
Екран роботи програми показаний на рис. Б.2.

Завдання 4. Організація меню.

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

“Номер завдання:” – введення номеру завдання.

Вихідні дані (ім'я, опис, тип):

Перевірка високосного року — якщо ввели число «1»

Розрахунок параметрів коричневої області — якщо ввели число «2».

На рис. 3 показаний приклад діаграми для завдання 4

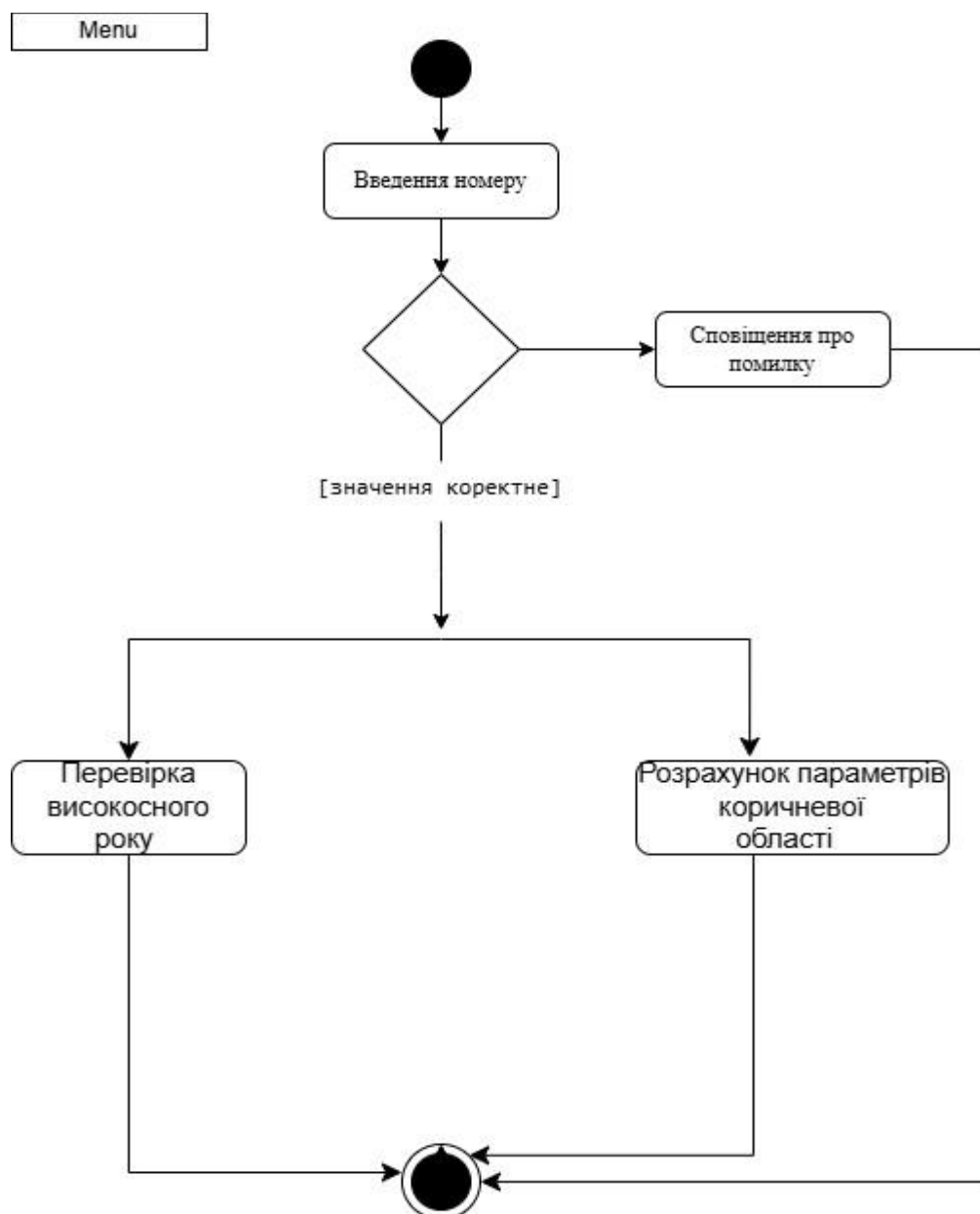


Рисунок 3 – Діаграма для завдання 4

ВИСНОВКИ

Розглянуто синтаксис мови C++ та визначено практичне представлення алгоритмів з розгалуженням у вигляді діаграм активності UML.

Розроблено структуру програми з функціями та доступом до меню.

ДОДАТОК А

Лістинг коду програми

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

// Перевірка високосного року
bool isLeapYear(int year) {
    if ((year % 4 == 0 && year % 100 != 0) || (year % 400 == 0)) {
        return true;
    }
    return false;
}

void checkLeapYear() {
    std::cout << "\nПеревірка високосного року\n";
    std::cout << "Введіть номер року: ";
    int year;
    std::cin >> year;

    if (year <= 0) {
        std::cerr << "Рік має бути додатнім числом.\n";
        return;
    }

    if (isLeapYear(year)) {
        std::cout << "Рік " << year << " є високосним і має 366 днів.\n";
    } else {
        std::cout << "Рік " << year << " є звичайним і має 365 днів.\n";
    }
}

// Функція перевірки точки в коричневій області
bool isPointInBrownArea(double x, double y, double a) {
    double r = a / 2.0; // Радіус кола

    // Перевірка для кожної чверті
    bool inFirstQuarter = (x >= 0 && y >= 0 && x * x + y * y <= r * r && y > -x + r);
    bool inSecondQuarter = (x <= 0 && y >= 0 && x * x + y * y <= r * r && y > x + r);
    bool inThirdQuarter = (x <= 0 && y <= 0 && x * x + y * y <= r * r && y >= -x -
r);
    bool inFourthQuarter = (x >= 0 && y <= 0 && x * x + y * y <= r * r && y >= x -
r);

    return inFirstQuarter || inSecondQuarter || inThirdQuarter || inFourthQuarter;
}

// Розрахунок площі та периметра коричневої області
void calculateBrownArea() {
    std::cout << "\nРозрахунок параметрів коричневої області\n";
    std::cout << "Введіть сторону квадрата a: ";
    double a;
    std::cin >> a;
```

```

if (a <= 0) {
    std::cerr << "Сторона квадрата має бути додатнім числом.\n";
    return;
}

double r = a / 2.0;

// Розрахунок площі коричневої області
double sectorArea = (M_PI * r * r) / 4.0; // Площа сектора кола
double triangleArea = (a * a) / 8.0;      // Площа трикутника
double brownArea = sectorArea - triangleArea;

// Розрахунок периметра коричневої області
double arcLength = (M_PI * r) / 2.0;      // Довжина дуги сектора
double brownPerimeter = arcLength + 2 * r; // Дуга + 2 радіуси

std::cout << "Площа коричневої області: " << brownArea << " квадратних
одиниць.\n";
std::cout << "Периметр коричневої області: " << brownPerimeter << " одиниць.\n";

// Перевірка точки
std::cout << "\nВведіть координати точки (x, y): ";
double x, y;
std::cin >> x >> y;

if (isPointInBrownArea(x, y, a)) {
    std::cout << "Точка (" << x << ", " << y << ") належить коричневій
області.\n";
} else {
    std::cout << "Точка (" << x << ", " << y << ") не належить коричневій
області.\n";
}
}

int main() {
    std::cout << "Оберіть завдання:\n";
    std::cout << "1 - Перевірка високосного року\n";
    std::cout << "2 - Розрахунок параметрів коричневої області\n";
    std::cout << "Ваш вибір: ";

    int choice;
    std::cin >> choice;

    switch (choice) {
        case 1:
            checkLeapYear();
            break;
        case 2:
            calculateBrownArea();
            break;
        default:
            std::cerr << "Неправильний вибір. Завершення програми.\n";
    }

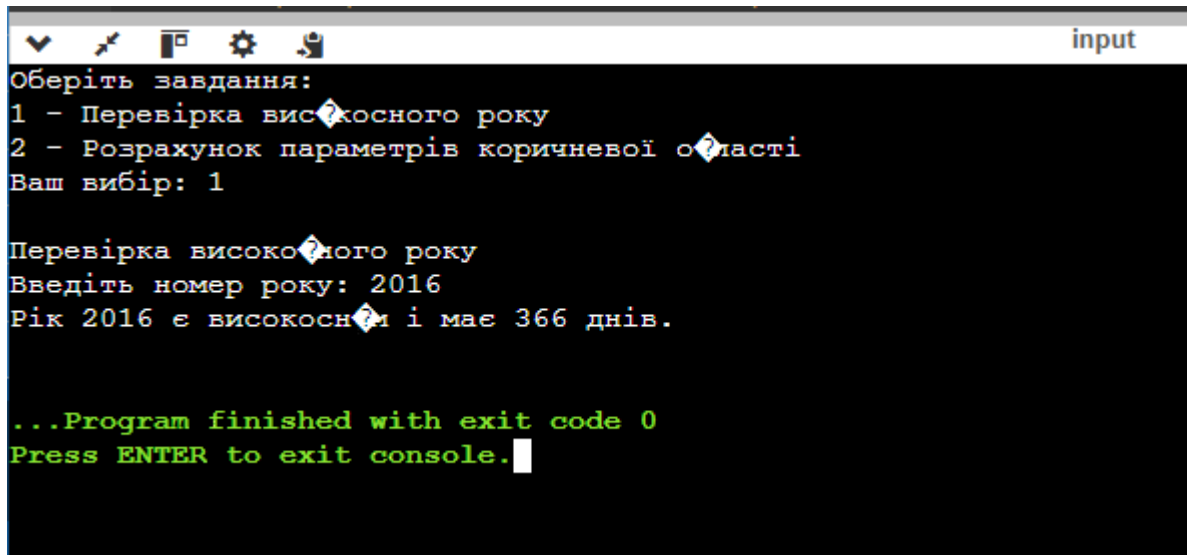
    return 0;
}

```


ДОДАТОК Б

Скрін-шоти вікна виконання програми

На рис. Б.1 показаний приклад виконання завдання 1.



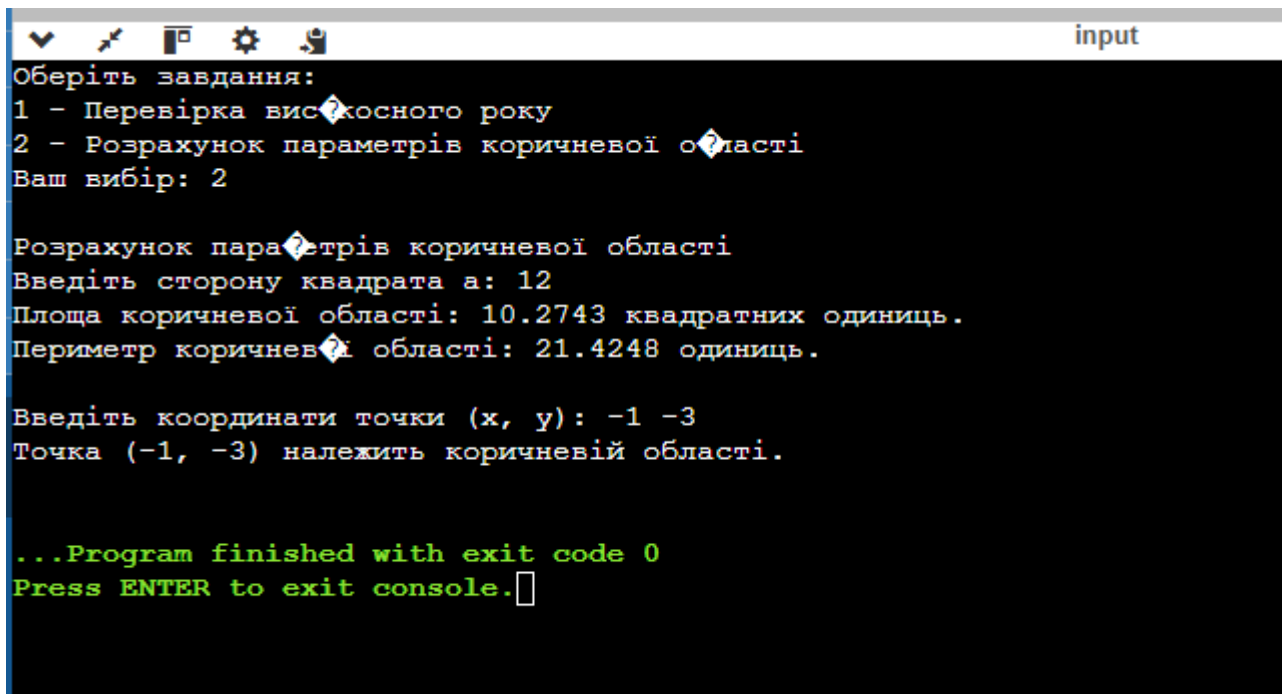
```
input
Оберіть завдання:
1 - Перевірка високосного року
2 - Розрахунок параметрів коричневої області
Ваш вибір: 1

Перевірка високосного року
Введіть номер року: 2016
Рік 2016 є високосним і має 366 днів.

...Program finished with exit code 0
Press ENTER to exit console.
```

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

На рис. Б.2 показаний приклад виконання завдання 2.



```
input
Оберіть завдання:
1 - Перевірка високосного року
2 - Розрахунок параметрів коричневої області
Ваш вибір: 2

Розрахунок параметрів коричневої області
Введіть сторону квадрата a: 12
Площа коричневої області: 10.2743 квадратних одиниць.
Периметр коричневої області: 21.4248 одиниць.

Введіть координати точки (x, y): -1 -3
Точка (-1, -3) належить коричневій області.

...Program finished with exit code 0
Press ENTER to exit console.
```

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