

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.301.175.318.4 ЛР

Виконав студент гр. 318

Єлизавета ВЕРБИЦЬКА

(підпис, дата)

(П.І.Б.)

Перевірів

 к.т.н., доц. Олена ГАВРИЛЕНКО

(підпис, дата)

(П.І.Б.)

2024

МЕТА РОБОТИ

Вивчити 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. (1f 30)

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. (8 варіант)

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

Завдання 3. (8 варіант)

Обчислити площу і периметр плоскої фігури.

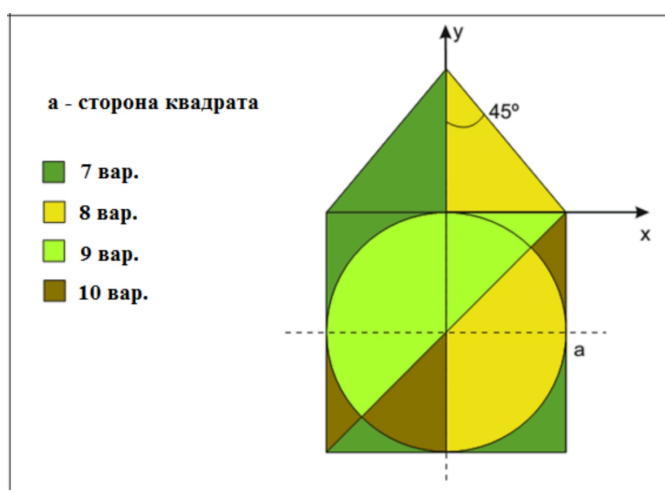


Рисунок 1 – умова 8 варіанта

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 1 (If30)

Вхідні дані:

number - будь-яке ціле число

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

Опис введеного числа

Алгоритм:

- 1) Введення запрошення до вводу
- 2) Введення значення змінної number
- 3) Виведення опису введеного числа

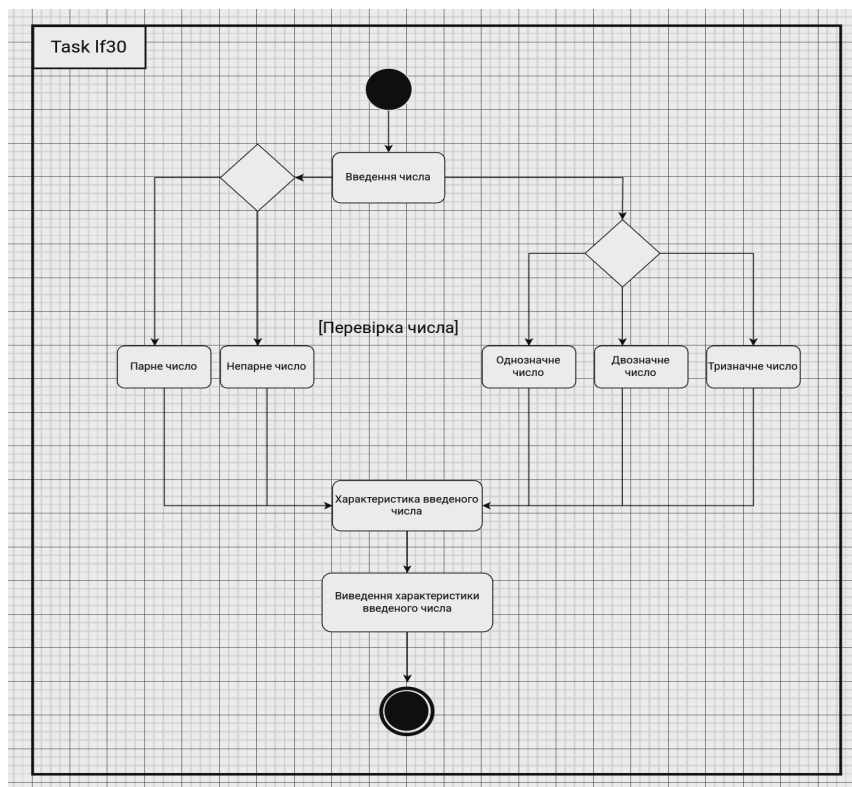


Рисунок 2 – Діаграма для завдання If 30

Лістинг коду вирішення задачі If30 наведено в дод. А . Екран роботи програми показаний на дод. Б на рис.

Екран роботи програми показаний на рис. Б.1

Вирішення задачі 2 (geom 8)

Вхідні дані:

a - змінна, значення сторони a

Вихідні дані:

triangle - значення площі трикутника

circle - значення площі кола

Алгоритм:

- 1) Введення запрошення до вводу
- 2) Введення значення сторони a
- 3) Розрахунок площі трикутника
- 4) Розрахунок площі кола
- 5) Виведення площ фігур

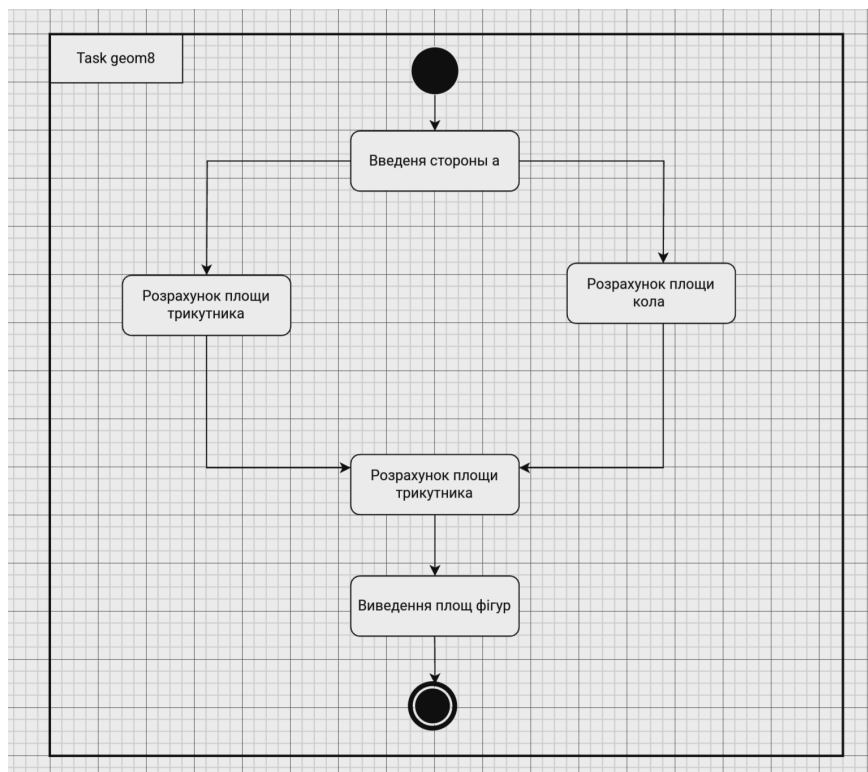


Рисунок 3 – Діаграма для завдання geom 8

Лістинг коду вирішення задачі номер 2 (8 варіант) в дод. А . Екран роботи програми показаний на дод. Б на рис.

Екран роботи програми показаний на рис. Б.2

Вирішення задачі номер 3 (geom 8)

Вхідні дані:

x - змінна, координати точки по осі x

y - змінна, координати точки по осі y

a - змінна, сторона фігури

Вихідні дані:

Виведення тексту чи належить точка одній із фігур

Алгоритм:

- 1) Запрошення до вводу
- 2) Введення даних змінних x, y, a
- 3) Розрахунок даних
- 4) Виведення результату

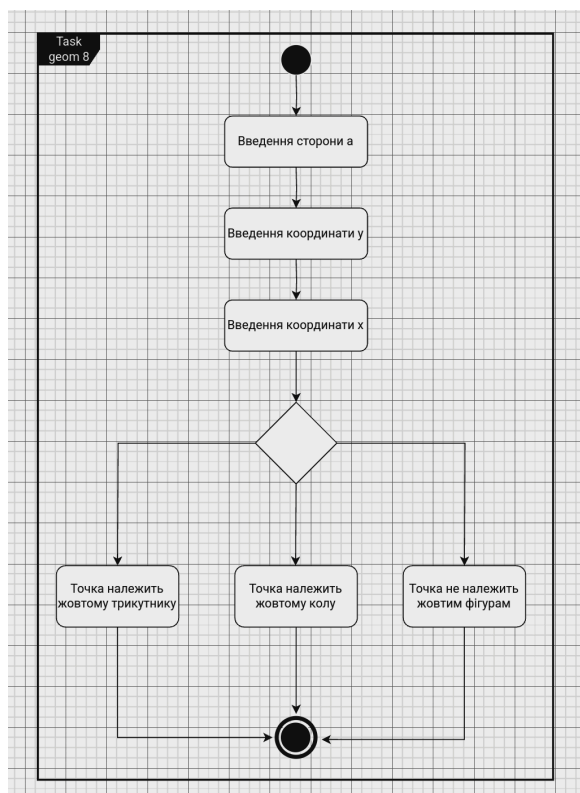


Рисунок 4 – Діаграма для завдання geom 8

Лістинг коду вирішення задачі номер 3 (8 варіант) наведено в дод. А . Екран роботи програми показаний на дод. Б на рис.

Екран роботи програми показаний на рис. Б.3

Вирішення задачі 4

Вхідні дані:

task - змінна

Вихідні дані:

В залежності від значення змінної task запуск одної з трьох задач

Алгоритм

- 1) Запрошення до вводу
- 2) Введення змінної a
- 3) Проігрування однієї з трьох програм в залежності від значення змінної a
- 4) Розрахунок виведеної задачі
- 5) Виведення результату

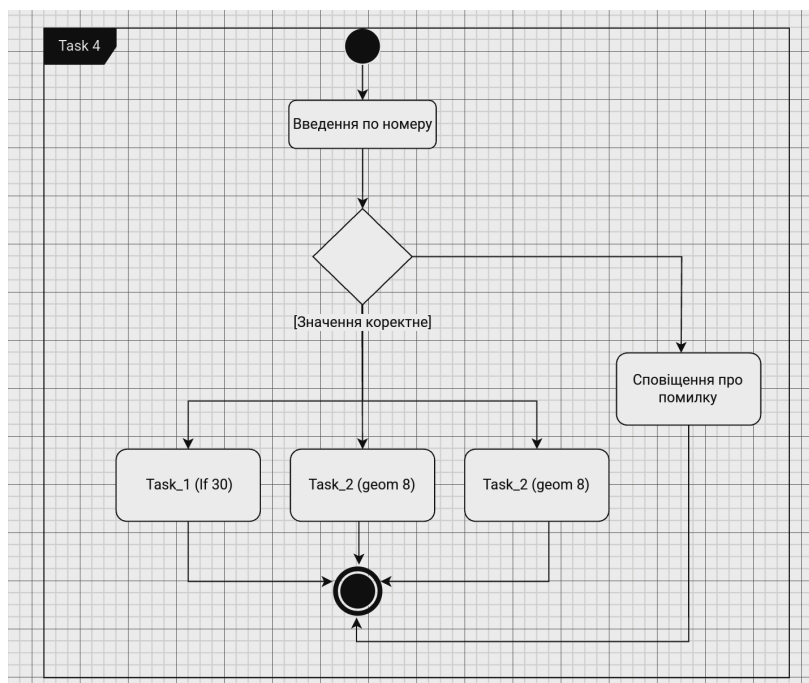


Рисунок 5 – Діаграма для завдання 4

Лістинг коду вирішення задачі 4 наведено в дод. А . Екран роботи програми показаний на дод. Б на рис.

Екран роботи програми показаний на рис. Б.4

ВИСНОВКИ

Вивчено матеріал щодо синтаксису у мові C++ і подання у вигляді UML діаграм активності алгоритмів з розгалуженням. Було відпрацьовано структурування програми з функціями і викликом меню.

ДОДАТОК А

Лістинг коду програми

Завдання 1 (1f30)

```
#include <iostream>
using namespace std;

int main() {

    int number; // Змінна для збереження числа, яке введе користувач

    cout << "Введіть число від 1 до 999: "; // Запит числа у користувача

    cin >> number; // Зчитування введеного значення

    // Перевіряємо, чи число у межах 1-999
    if (number < 1 || number > 999) {
        cout << "Число поза діапазоном 1-999." << endl; // Повідомлення про
невідповідність
        return 1; // Завершення програми, якщо число неправильне
    }

    // Визначаємо, парне чи непарне число
    string parity = (number % 2 == 0) ? "парне" : "непарне";

    string digits; // Змінна для опису кількості цифр у числі

    // Визначення, скільки цифр у числі
    if (number < 10)
        digits = "однозначне"; // Число менше 10 — це однозначне число
    else if (number < 100)
        digits = "двозначне"; // Число менше 100 — це двозначне
```

```

else
    digits = "тризначне"; // Інакше – тризначне

// Вивід результатів: парність і кількість цифр
cout << parity << " " << digits << " число" << endl;

return 0;
}

```

Завдання 2 (8 варіант)

```

#include <iostream>
#include <cmath>

// Обчислює площу трикутника, який вписаний у квадрат зі стороною a
double calculatetriangle(double a) {
    return a * a / 8.0;
}

// Обчислює площу півкола, яке вписане в квадрат зі стороною a
double calculatecircle(double a) {
    return (M_PI * a * a) / 16.0;
}

int main() {
    double a; // Змінна для введення сторони квадрата
    std::cout << "Введіть сторону a: "; // Виведення підказки для вводу
    std::cin >> a; // Зчитування значення сторони a

    // Обчислення площ
    double triangle = calculatetriangle(a); // Виклик функції для площі трикутника
    double circle = calculatecircle(a);    // Виклик функції для площі півкола

    // Виведення результатів
    std::cout << "Площа жовтого трикутника: " << triangle << std::endl;
    std::cout << "Площа жовтого півкола: " << circle << std::endl;

    return 0; // Завершення програми
}

```

Завдання 3 (8 варіант)

```

#include <iostream>

```



```

#include <cmath>

// Перевіряє, чи знаходиться точка у жовтому трикутнику
bool Вжовтомутрикутнику(double x, double y, double a) {
    return (y >= x) && (y <= a) && (x <= a / 2);
}

// Перевіряє, чи точка належить жовтому півколу
bool Вжовтомуполукрузі(double x, double y, double a) {
    double centerX = a / 2;
    double centerY = a / 2;
    double radius = a / 2;

    // Відстань від точки до центра кола в квадраті
    double distanceSquared = (x - centerX) * (x - centerX) + (y - centerY) * (y -
centerY);
    return (distanceSquared <= radius * radius) && (x >= centerX);
}

int main() {
    double x, y, a; // Змінні для введення
    std::cout << "Введіть сторону квадрата a: "; // Запит розміру квадрата
    std::cin >> a; // Зчитування сторони квадрата
    std::cout << "Введіть координати точки (x y): "; // Запит координат точки
    std::cin >> x >> y; // Зчитування координат

    // Виклики функцій для перевірки розташування точки
    bool Втрикутнику = Вжовтомутрикутнику(x, y, a);
    bool Вполукрузі = Вжовтомуполукрузі(x, y, a);

    // Вивід результатів
    if (Втрикутнику) {
        std::cout << "Точка належить жовтому трикутнику." << std::endl;
    } else if (Вполукрузі) {
        std::cout << "Точка належить жовтому колу." << std::endl;
    } else {
        std::cout << "Точка не належить жовтим фігурам." << std::endl;
    }

    return 0; // Завершення програми
}

```

Завдання 4

```
#include <iostream>
#include <cmath>
using namespace std;

// Функція для обчислення площі жовтого півкола
double calculatecircle(double a) {
    return (M_PI * a * a) / 16.0;
}

// Функція для обчислення площі жовтого трикутника
double calculatetriangle(double a) {
    return a * a / 8.0;
}

// Перевірка, чи знаходиться точка у жовтому трикутнику
bool Вжовтомутрикутнику(double x, double y, double a) {
    return (y >= x) && (y <= a) && (x <= a / 2);
}

// Перевірка, чи знаходиться точка у жовтому півколі
bool Вжовтомуполукрузі(double x, double y, double a) {
    double centerX = a / 2;
    double centerY = a / 2;
    double radius = a / 2;

    double distanceSquared = (x - centerX) * (x - centerX) + (y - centerY) * (y - centerY);
    return (distanceSquared <= radius * radius) && (x >= centerX);
}
```

```

int main() {
    int task; // Змінна для вибору пункту меню

    while (true) { // Цикл працює без виходу
        // Виводимо меню
        cout << "\nМеню:\n";
        cout << "1: Число від 1 до 999\n";
        cout << "2: Площа жовтих фігур\n";
        cout << "3: Перевірка точки у фігурах\n";
        cout << "Введіть номер завдання: ";
        cin >> task; // Зчитуємо вибір користувача

        switch (task) {
            case 1: { // Завдання 1: Перевірка числа
                int number;
                cout << "Введіть число від 1 до 999: ";
                cin >> number;

                if (number < 1 || number > 999) {
                    cout << "Число поза діапазоном 1-999." << endl;
                } else {
                    string parity = (number % 2 == 0) ? "парне" : "непарне";
                    string digits;
                    if (number < 10)
                        digits = "однозначне";
                    else if (number < 100)
                        digits = "двозначне";
                    else
                        digits = "тризначне";

                    cout << parity << " " << digits << " число" << endl;
                }
                break;
            }
        }
    }
}

```

```

case 2: { // Завдання 2: Розрахунок площ
    double a;
    cout << "Введіть сторону a: ";
    cin >> a;

    double triangle = calculatetriangle(a);
    double circle = calculatecircle(a);

    cout << "Площа жовтого трикутника: " << triangle << endl;
    cout << "Площа жовтого півкола: " << circle << endl;
    break;
}
case 3: { // Завдання 3: точки у фігурах
    double x, y, a;
    cout << "Введіть сторону квадрата a: ";
    cin >> a;
    cout << "Введіть координати точки (x, y): ";
    cin >> x >> y;

    bool Втрикутнику = Вжовтомутрикутнику(x, y, a);
    bool Вполукрузі = Вжовтомуполукрузі(x, y, a);

    if (Втрикутнику) {
        cout << "Точка належить жовтому трикутнику." << endl;
    } else if (Вполукрузі) {
        cout << "Точка належить жовтому півколу." << endl;
    } else {
        cout << "Точка не належить жовтим фігурам." << endl;
    }
    break;
}
default: { // Некоректний вибір
    cout << "Некоректний вибір. Спробуйте ще раз." << endl;
    break;
}

```

```
    }  
}  
  
return 0;  
}
```

ДОДАТОК Б

Скріншоти вікна виконання програми

```
Введіть число від 1 до 999: 123  
непарне тризначне число  
  
[Program finished]
```

Рисунок Б.1 – Екран виконання програми для вирішення завдання
Завдання 1 (lf30)

```
Введіть сторону а: 21  
Площа жовтого трикутника: 55.125  
Площа жовтого півкола: 86.5901  
  
[Program finished]
```

Рисунок Б.2 – Екран виконання програми для вирішення завдання
Завдання 2 (geom 8)

```
Введіть сторону квадрата a: 10
Введіть координати точки (x y): 3 8
Точка належить жовтому трикутнику.

[Program finished]
```

Рисунок Б.3 – Екран виконання програми для вирішення завдання
Завдання 3 (geom 8)

```
Оберіть завдання:
1: Число від 1 до 999;
2: Площа жовтих фігур;
3: Перевірка точки у фігурах.
Введіть номер завдання: 1
Введіть число від 1 до 999: 150
парне тризначне число

[Program finished]
```

Рисунок Б.4 – Екран виконання програми для вирішення завдання
Завдання 4