

Міністерство освіти і науки України  
Центральноукраїнський національний технічний університет  
Механіко-технологічний факультет

ЗВІТ  
ПРО ВИКОНАННЯ ЛАБОРАТОРНОЇ РОБОТИ № 12  
з навчальної дисципліни  
“Базові методології та технології програмування”  
ПРОГРАМНА РЕАЛІЗАЦІЯ ОБРОБЛЕННЯ МАСИВІВ  
ДАНИХ ТА СИМВОЛЬНОЇ ІНФОРМАЦІЇ

ЗАВДАННЯ ВИДАВ  
доцент кафедри кібербезпеки  
та програмного забезпечення  
Доренський О. П.  
<https://github.com/odorenskyi/>

ВИКОНАВ  
студент академічної групи КБ-23  
Чернолес К.С.

ПЕРЕВІРИВ ст. викладач  
кафедри кібербезпеки  
та програмного забезпечення

Дрєєва Г.М.



*Мета роботи* : полягає у набутті ґрунтовних вмінь і практичних навичок реалізації технології модульного програмування, застосування операторів C/C++ арифметичних, логічних, побітових операцій, умови, циклів та вибору під час розроблення статичних бібліотек, заголовкових файлів та програмних засобів у кросплатформовому середовищі Code::Blocks.

1. Як складову заголовкового файлу ModulesПрізвище.h розробити клас ClassLab12\_Прізвище - формальне представлення абстракції сутності предметної області (об'єкта) за варіантом, - поведінка об'єкта якого реалізовує розв'язування задачі 7.1.

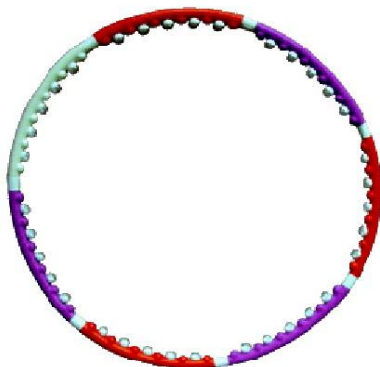
2. Реалізувати додаток Teacher, якій видає 100 звукових сигналів і в текстовий файл TestResults.txt запитує рядок "Встановлені вимоги порядку виконання лабораторної роботи порушено!", якщо файл проекту main.cpp під час його компіляції знаходиться не в \Lab12\prj, інакше - створює об'єкт класу ClassLab12\_Прізвище із заголовкового файлу ModulesПрізвище.h та виконує його unit-тестування за тест-сьютом(ами) із \Lab12\TestSuite\, протоколюючи результати тестування в текстовий файл \Lab12\TestSuite\TestResults.txt.

Варіант 5

## ВАРІАНТ № 5

## — ЗАДАЧА 12.1 —

Дано наступну сутність предметної області (об'єкт).



Об'єкт<sup>1</sup> (екземпляр) класу `ClassLab12_Прізвище`, як абстракція даної сутності предметної області, за наданим інтерфейсом забезпечує:

- надання<sup>2</sup> значень своїх атрибутів;
- надання значення своєї площі<sup>3</sup>;
- зміну значення заданого атрибута(ів)<sup>4</sup>.

<sup>1</sup> Під час створення об'єкта класу всі його атрибути ініціалізуються конструктором.

<sup>2</sup> Під наданням розуміється повернення результату відповідними функціями-членами об'єкта класу.

<sup>3</sup> Площа обчислюється і повертається відповідною функцією-членом (методом) об'єкта класу за значеннями його атрибутів.

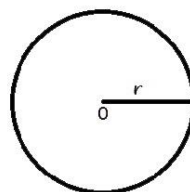
<sup>4</sup> Всі дані-члени класу є закритими (`private`); доступ до них (читання, запис) реалізують відповідні відкриті функції-члени (`public`), які у свою чергу забезпечують валідацію вхідних даних.



Площу кола можна обчислити:

$$S = \pi r^2 \quad \text{або} \quad S = \frac{1}{4} \pi d^2,$$

де  $S$  – площа кола,  $R$  – радіус,  $d$  – діаметр.



Аналіз лабораторної роботи:

Ця програма написана на мові C++ і демонструє використання класів, методів, а також основних операцій з атрибутами об'єктів. Програма містить визначення класу `ClassLab12_Chernoles`, який моделює коло з атрибутами радіуса та кольору. Вона також демонструє основні методи класу для доступу до атрибутів, їх зміни та розрахунку площі кола.

**Лістинг програмного коду:**

```
#include <iostream>

#include <string>

#include <cmath>

class ClassLab12_Chernoles {
private:
    double radius;
    std::string color;

public:
    // Конструктор
    ClassLab12_Chernoles(double r, std::string c) : radius(r), color(c) {}

    // Сеттери
    void setRadius(double r) {
        if (r > 0) {
            radius = r;
        }
    }

    void setColor(const std::string& c) {
        color = c;
    }

    // Геттери
    double getRadius() const {
        return radius;
    }

    std::string getColor() const {
        return color;
    }

    // Метод для обчислення площі
    double calculateArea() const {
        return M_PI * radius * radius;
    }
}
```

```

// Метод для зміни площі та оновлення радіуса
void setArea(double area) {
    if (area > 0) {
        radius = std::sqrt(area / M_PI);
    }
}

// Метод для зміни значення заданого атрибуту
void updateAttribute(const std::string& attributeName, const std::string&
value) {
    if (attributeName == "color") {
        setColor(value);
    } else if (attributeName == "radius") {
        try {
            double r = std::stod(value);
            setRadius(r);
        } catch (const std::invalid_argument& e) {
            std::cerr << "Invalid radius value: " << value << std::endl;
        }
    } else if (attributeName == "area") {
        try {
            double area = std::stod(value);
            setArea(area);
        } catch (const std::invalid_argument& e) {
            std::cerr << "Invalid area value: " << value << std::endl;
        }
    }
}

};

int main() {
    double radius;
    std::string color;

    // Введення даних користувачем
    std::cout << "Enter radius: ";
    std::cin >> radius;
    std::cin.ignore(); // Щоб ігнорувати залишений символ нового рядка
    std::cout << "Enter color: ";
    std::getline(std::cin, color);

```

```

// Створення об'єкта класу з введеними даними
ClassLab12_Chernoles obj(radius, color);

// Вивід значень атрибутів
std::cout << "Radius: " << obj.getRadius() << std::endl;
std::cout << "Color: " << obj.getColor() << std::endl;

// Обчислення та вивід площі
std::cout << "Area: " << obj.calculateArea() << std::endl;

return 0;
}

```

**Висновок:** Ця програма ілюструє основи роботи з класами у C++, включаючи використання конструкторів, сеттерів, геттерів, обчислювальних методів і методів для оновлення атрибутів. Вона також демонструє обробку помилок при некоректному введенні даних для атрибута radius.

Назва тестового набору Test Suite Description	TS_12_1
Назва проекту / ПЗ Name of Project / Software	Lab #12
Рівень тестування Level of Testing	модульний
Автор тест-сьюта Test Suite Author	Чернолєс Кирило
Виконавець Implementer	Чернолєс Кирило

[Розрив обтікання текстом]

Ід-р тест- кейса / Test Case ID	Дії (кроки) / Action (Test Steps)	Очікуваний результат / Expected Result	Результат тестування (пройшов/не вдалося/ заблокований) / Test Result (passed/failed/ blocked)
TCM_1	Вхід: Radius: 20 Color: red	Вихід: Radius: 20 Color: red Area: 1256.64	PASSED

TCM_2	Вхід: Radius:1 Color:blue	Вихід: Radius: 1 Color: blue Area: 3.14159	PASSED
TCM_3	Вхід: Radius: 0 Color: red	Вихід: Radius: 0 Color: red Area: 0	PASSED
TCM_4	Вхід: Radius: -1 Color: green	Вихід: Radius: -1 Color: green Area: 3.14159	PASSED
TCM_5	Вхід: Enter radius:1000000000 Enter color:blue	Вихід: Radius: 1e+09 Color: blue Area: 3.14159e+18	PASSED