

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 АБСТРАКТНИХ ТИПІВ ДАНИХ

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

ВИКОНАВ
студент академічної групи
КІ-23
Чепіль В.О.

ПЕРЕВІРИВ
ст. викладач
кафедри кібербезпеки
та програмного забезпечення
Дресва Г. М.

Тема: Програмна реалізація абстрактних типів даних

Мета роботи: Набуття ґрунтовних вмінь і практичних навичок об'єктного аналізу й проектування, створення класів C++ та тестування їх екземплярів, використання препроцесорних директив, макросів і макрооператорів під час реалізації програмних засобів у кросплатформовому середовищі Code::Blocks.

Завдання до лабораторної роботи:

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, який видає 100 звукових сигналів і в текстовий файл TestResults.txt записує рядок “Встановлені вимоги порядку виконання лабораторної роботи порушено!”, якщо файл проекту main.cpp під час його компіляції знаходився 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№2

ВАРІАНТ № 2

— ЗАДАЧА 12.1 —

Дано наступну сутність предметної області (об'єкт).



Об'єкт¹ (екземпляр) класу `ClassLab12_Прізвище`, як абстракція даної сутності предметної області, за наданим інтерфейсом забезпечує:

- надання² значень своїх атрибутів;
- надання значення свого об'єму³;
- зміну значення заданого атрибута(ів)⁴.

¹ Під час створення об'єкта класу всі його атрибути ініціалізуються конструктором.

² Під наданням розуміється повернення результату відповідними функціями-членами об'єкта класу.

³ Об'єм обчислюється і повертається відповідною функцією-членом (методом) об'єкта класу за значеннями його атрибутів.

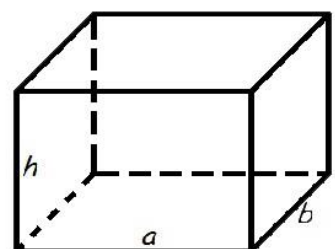
⁴ Всі дані-члени класу є закритими (`private`); доступ до них (читання, запис) реалізують відповідні відкриті функції-члени (`public`), які у свою чергу забезпечують валідацію вхідних даних.



Об'єм прямокутного паралелепіпеда рівний добутку його довжини, ширини і висоти:

$$V = a \cdot b \cdot h,$$

де V – об'єм прямокутного паралелепіпеда,
 a – довжина, b – ширина, h – висота.



Аналіз та постановка задачі 12.1

Концептуалізація предметної області

Об'єктом, згідно з варіантом завдання, є акваріум.

Об'єктний аналіз

Атрибути:

- Довжина: довжина акваріуму.
- Ширина: ширина акваріуму.
- Висота: висота акваріуму.

Визначення інтерфейсів сутності

Отримання та задання значення: довжини, ширини, висоти акваріуму. Прототипи функції будуть визначені при проектуванні модуля та класа.

Аналіз вимог до програмного модуля ModulesChepil

Назва класу - ClassLab12_Chepil. Атрибути ініціалізуються конструктором. Об'єм паралелепіпеда (акваріума) обчислюється і повертається відповідною функцією-членом за значенням атрибутів. Доступ до даних-членів повинен бути закритим (private), для атрибутів повинні бути відповідні відкриті (public) функції-члени (читання та запис значення), що забезпечують валідацію вхідних даних.

Для атрибутів будуть використовуватись такі типи даних:

- length — довжина (float)
- width — ширина (float)
- height — висота (float)

Приклад формату тест-кейса

30;50;20;

54;62;78;

254;3;78;

2;78;34;

0;0;0;

Процедура читання тест-кейсів:

Відкриття файлу TestSuite.txt для читання:

Використовується стандартна бібліотека fstream для роботи з файлами. Файл відкривається для читання за допомогою ifstream.

Читання рядків з файлу:

Кожен рядок представляє окремий тест-кейс. Рядки читаються по черзі за допомогою функції getline.

Розділення рядка на поля:

Кожен рядок розділяється на частини за допомогою символу ";".

Для цього використовується об'єкт stringstream.

Створення об'єкта класу ClassLab12_Chepil.

Запис результатів тестування у файл TestResults.txt:

Результати кожного тест-кейса записуються у файл TestResults.txt.

Формат запису результатів включає всі властивості акваріуму (довжину, ширину, висоту та об'єм).

Лістинг ModulesChepil.h:

```
#ifndef MODULESCHEPIL_H_INCLUDED
#define MODULESCHEPIL_H_INCLUDED
```

```
class ClassLab12_Chepil
```

```
{
```

```
private:
```

```
    float length;
```

```
    float width;
```

```
    float height;
```

```
    void valid(float value) {
```

```
        if (value <= 0) {
```

```
            throw std::invalid_argument("Значення має бути позитивним.");
```

```

    }
}

public:
    ClassLab12_Chepil(float len, float wid, float hei) {
        valid(len);
        valid(wid);
        valid(hei);
        length = len;
        width = wid;
        height = hei;
    }

    float getLength() const {
        return length;
    }

    float getWidth() const {
        return width;
    }

    float getHeight() const {
        return height;
    }

    void setLength(float len) {
        valid(len);
        length = len;
    }

    void setWidth(float wid) {
        valid(wid);
        width = wid;
    }

    void setHeight(float hei) {
        valid(hei);
        height = hei;
    }

    float getVolume() const {
        return length * width * height;
    }
};

#endif // MODULESCHEPIL_H_INCLUDED

```

Лістинг додатка Teacher:

```
#include <iostream>
#include <Windows.h>
#include <fstream>
#include <sstream>

#include "ModulesChepil.h"

#define TEST_SUITE_FILE "\\lab12\\TestSuite\\TestSuite.txt"
#define TEST_RESULTS_FILE "\\lab12\\TestSuite\\TestResults.txt"

using namespace std;

int main()
{
    system("chcp 65001 & cls");

    string currentFilePath = __FILE__;
    int checkResult = currentFilePath.find("\\lab12\\prj\\");
    if (checkResult == -1) {
        for (int i = 0; i < 100; ++i) {
            Beep(500, 100);
        }

        ofstream resultFile(TEST_RESULTS_FILE);
        if (resultFile.is_open()) {
            resultFile << "Встановлені вимоги порядку виконання лабораторної роботи  
порушено!";
            resultFile.close();
        } else {
            cerr << "Не вдалося відкрити файл для запису.";
            return 1;
        }
    } else {
        ifstream inpFile(TEST_SUITE_FILE);
        ofstream outFile(TEST_RESULTS_FILE);
        string line;
        int testCase = 0;

        while (getline(inpFile, line)) {
            stringstream ss(line);
            float length;
            float width;
            float height;
            ss >> length;
            ss.ignore(1, ';');
            ss >> width;
            ss.ignore(1, ';');
            ss >> height;
            ss.ignore(1, ';');

            ClassLab12_Chepil testObj = ClassLab12_Chepil(length, width, height);
```

```

        testCase += 1;
        outFile << "Тест " << testCase << endl;
        outFile << "Довжина: " << testObj.getLength() << endl;
        outFile << "Ширина: " << testObj.getWidth() << endl;
        outFile << "Висота: " << testObj.getHeight() << endl;
        outFile << "Об'єм: " << testObj.getVolume() << endl << endl;
    }

    cout << "Роботу застосунку завершено, результат збережений у файлі TestResults.txt"
    << endl;
    system("pause");

    inpFile.close();
    outFile.close();
}

return 0;
}

```

Висновок: Виконуючи цю лабораторну роботу, я здобув глибокі знання та практичний досвід у програмній реалізації абстрактних типів даних на мові програмування C++. Особлива увага приділялась роботі з файлами, зокрема перевірці розташування файлів проекту, запису та читанню даних, а також реалізації та тестуванню класів.

Згідно з методичними рекомендаціями, необхідно було послідовно проаналізувати та виконати задачі 12.1 та 12.2, вивчити вимоги до програмного забезпечення та змісту вхідного файлу, спроектувати архітектуру класу та задокументувати результати у звіті. У цій лабораторній роботі особливий акцент робився на перевірці правильного розташування файлів проекту та протоколюванні результатів тестування.

Задача 12.1

Для виконання задачі 12.1 було створено клас ClassLab12_Cheril, який представляє об'єкт "акваріум". Клас включає атрибути довжина, ширина, висота, а також методи для їхнього задання і отримання.

Обчислення об'єму акваріума здійснюється за допомогою відповідної функції-члена класу.

Задача 12.2

Завдання полягало у створенні програми Teacher, яка перевіряє місцезнаходження файлу проекту main.cpp під час компіляції. Якщо файл знаходився не у вказаному каталозі, програма генерувала 100 звукових сигналів та записувала повідомлення про порушення вимог у текстовий файл. Якщо файл знаходився у правильному каталозі, додаток створював об'єкт класу ClassLab12_Chepil та виконував його unit-тестування за допомогою тест-сьюту, результати якого записувалися у текстовий файл TestResults.txt.

Результати тестування

Був реалізований тестовий драйвер, який протоколював процес тестування. Було розроблено кілька тестових кейсів, що включали різні комбінації значень атрибутів акваріума. Усі тестові кейси успішно завершилися зі статусом "passed".

Відповіді на контрольні запитання:

1. Що є результатами виконання концептуалізації предметної області, об'єктного аналізу та визначення інтерфейсів сутностей предметної області?

Концептуалізація: визначення сутностей, їх властивостей та зв'язків.

Об'єктний аналіз: детальна модель класів та об'єктів.

Визначення інтерфейсів: специфікація методів і взаємодій між сутностями.

2. Який зв'язок між процесом концептуалізації предметної області та процесами об'єктного аналізу і визначення інтерфейсів?

Концептуалізація створює загальну картину сутностей.

Об'єктний аналіз деталізує структуру і поведінку сутностей.

Визначення інтерфейсів встановлює способи взаємодії між об'єктами.

3. Сформулюйте критерії, за якими чітко можливо визначити: абстракцію сутності предметної області слід описати мовою C++ типом структура (struct) чи типом клас (class)?

struct: прості дані, всі члени публічні.

class: складніші абстракції, контроль доступу, інкапсуляція.

4. Що в програмуванні розуміють під інтерфейсом класу?

Набір публічних методів і властивостей, доступних для взаємодії з об'єктами класу.

5. Обґрунтовано поясніть, чому в класі C++ не можна оголосити конструктор з закритим рівнем доступу?

Можливий для шаблонів, як синглтон. В інших випадках обмежує створення об'єктів.

6. Здійсніть порівняльний аналіз перевантаженої функції та функції з параметрами за замовчуванням.

Перевантажені функції: однакове ім'я, різні параметри.

Параметри за замовчуванням: одна функція, параметри мають значення за замовчуванням.

7. За допомогою яких операторів C++ здійснюється доступ до відкритих членів об'єктів класу?

Крапка (.) для об'єктів.

Стрілка (->) для вказівників.

8. Яким чином клас C++ як абстрактний тип даних (ADT) дозволяє реалізувати принцип інкапсуляції?

Приховання деталей реалізації через специфікатори доступу (public, protected, private), контрольований доступ через методи.

