

Міністерство освіти і науки України
Центральноукраїнський національний технічний університет
Механіко-технологічний факультет
Кафедра кібербезпеки та програмного забезпечення

Звіт
Про виконання Лабораторії роботи №12
з дисципліни
“Базові методології та технології програмування”
на тему
«Програмна реалізація абстрактних типів даних»

Виконав
Студент академічної групи КН-22
Гончарук О.С.

Перевірив
викладач
Собінов О.Г

Тема: Програмна реалізація абстрактних типів даних.

Мета: полягає у набутті 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:

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 проектуа 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 12

ВАРІАНТ 13

— 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 площі своєї поверхні³;
- зміну значення заданого атрибута(`iv`)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 Площа поверхні обчислюється і повертається відповідною функцією-членом (методом) об'єкта класу за значеннями його атрибутів.

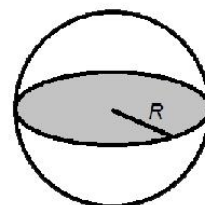
⁴ Всі дані-члени класу є закритими (`private`); доступ до них (читання, запис) реалізують відповідні відкриті функції-члени (`public`), які у свою чергу забезпечують валідацію вхідних даних.



Площу поверхні кулі можна обчислити:

$$S = 4\pi R^2 \quad \text{або} \quad S = \pi D^2,$$

де S – площа, R – радіус, D – діаметр.



Лістинг задач 12.1:

```
#ifndef MODULESGONCHARUK_H_INCLUDED
#define MODULESGONCHARUK_H_INCLUDED

#include <iostream>
#include <fstream>
#include <string>

void World()
{
    std::cout << "Wolrd!";
}

class ClassLab12_Goncharuk
{
public:
    ClassLab12_Goncharuk(const float radius = 1) :
        Radius(radius)
    {

    }
    float getS();
    float getRadius();
    float changeRadius(float radius);

private:
    float Radius;
};

float ClassLab12_Goncharuk::getS()
{
    return 4 * 3.14 * (Radius * Radius);
}

float ClassLab12_Goncharuk::changeRadius(float radius)
{
    Radius = radius;
}

float ClassLab12_Goncharuk::getRadius()
{
    return Radius;
}

bool checkFileInFolder()
{
    std::string file = __FILE__;
    size_t checking = file.find("\\prj");

    if (checking == std::string::npos)
    {
        std::ofstream fileResult("../TestSuite/TestResult.txt");

        fileResult << "Встановлені вимоги порядку виконання лабораторної роботи
порушено!\n";
        for (int i = 0; i < 100; i++)
            std::cout << "a";

        fileResult.close();
        return false;
    }
}
```

```

        return true;
    }

std::string writeResultToFile(ClassLab12_Goncharuk globus, int counter)
{
    return "TestSuite №" + std::to_string(counter)
        + "\radius = " + std::to_string(globus.getRadius())
        + "\nResult = " + std::to_string(globus.getS()) + "\n";
}

#endif // MODULESGONCHARUK_H_INCLUDED

```

Лістинг Teacher main.cpp:

```

#include <iostream>
#include <fstream>
#include <sstream>
#include "ModulesGoncharuk.h"

using namespace std;

int main()
{
    string line;
    if(checkFileInFolder() == false)
        return 0;

    ClassLab12_Goncharuk globus;
    ifstream Tests_File("../TestSuite/Tests.txt");
    ofstream TestResult_File("../TestSuite/TestResult.txt");

    if (Tests_File.is_open() || TestResult_File.is_open())
    {
        int counter = 1;
        float number;
        while (getline(Tests_File, line, '\n'))
        {
            istringstream iss(line);

            while (iss >> number)
                arrNumbers.push_back(number);

            globus.changeRadius(arrNumbers[0]);
            TestResult_File << writeResultToFile(globus, counter);
            counter++;
        }
    }
    Tests_File.close();
    TestResult_File.close();
    system("pause");
    return 0;
}

```

TestSuite

Назва тестового набору Test Suite Description	TestSuite
Назва проекту / ПЗ Name of Project / Software	Teacher.exe
Рівень тестування Level of Testing	модульне / Unit Testing
Автор тест-сьюта Test Suite Author	Гончарук Олександр Святославович

Test Case ID	Action	Expected Result	Test Result
TC-01	Radius = 4	S = 201.061932	passed
TC-02	Radius = 8	S = 804.247721	passed
TC-03	Radius = 12	S = 1809.557375	passed
TC-04	Radius = 16.5	S = 3421.194433	passed

Зміст вхідного файлу:

4
8
12
16.5

Зміст вихідного файлу:

Test Suite №1

Length = 4.000000

Result = 201.061932

Test Suite №2

Length = 8.000000

Result = 804.247721

Test Suite №3

Length = 12.000000

Result = 1809.557375

Test Suite №4

Length = 16.500000

Result = 3421.194433

Висновок

Під час виконання лабораторної роботи було зроблено дії:

Завантажено завдання із репозиторію. Після цього було створено папки prj, Software, TestSuite, та Report і заповнено README.md файл. Почергово було виконано аналіз і постановку задач 12.1 і 12.2, аналіз вимог до ПЗ та вмісту вхідного текстового файлу, проектування архітектури, детальне проектування програмних модулів; одержані артефакти задокументувати й включити до звіту. Далі було розроблено один тест-сьют задля проведення автоматизованого unit-тестування заголовкового файлу розв'язування задачі 12.1. Тестові артефакти було задокументовано та включено до звіту. За допомогою отриманих під час проектування програмних модулів артефактами виконати конструювання функцій: мовою програмування C++ було реалізовано функції, які реалізують розв'язування задач 12.1 і 12.2. Потім мовою програмування C++ було реалізовано консольний застосунок – тестовий драйвер для модульного тестування функцій розв'язування задач 12.1 за допомогою розробленого тест-сьюту з \Lab12\TestSuite та вхідного і/або вихідного текстового файлу. За допомогою Teacher.exe було виконано автоматизоване тестування розроблених функцій розв'язування задач 12.1 та 12.2. Вихідний код (текст) проектів Teacher та ModulesGoncharuk.h включено в звіт.

Під час цієї роботи я набув ґрунтовних вмінь і практичних навичок у об'єктному аналізі й проектуванні, створенні класів C++ та тестуванні їх екземплярів. Я також навчився використовувати препроцесорні директиви, макроси і макрооператори під час реалізації програмних засобів у кросплатформовому середовищі Code::Blocks. Під час створення класів C++ я навчився про об'єктно-орієнтоване програмування та його основні принципи, такі як інкапсуляція, спадкування та поліморфізм. Я також навчився проектувати класи, використовуючи діаграми класів та UML-нотації. Тестування екземплярів класів дозволяє перевірити їх функціональність та надійність перед їх використанням у реальній програмі. Ця робота дозволила мені отримати цінний досвід у програмуванні на мові C++, що буде корисним у моїй подальшій професійній діяльності.