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

ВИКОНАВ

студент академічної групи

КБ 22-2

\_\_\_\_\_\_\_\_\_ Ткаченко О. С.

ПЕРЕВІРИВ

викладач кафедри кібербезпеки

та програмного забезпечення

\_\_\_\_\_\_\_\_\_ Олександр СОБІНОВ

Кропивницький – 2023

**Мета роботи**

Полягає у набутті ґрунтовних вмінь і практичних навичок об’єктного аналізу й проектування, створення класів С++ 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 проекта main.срр 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 16**

****

Рисунок 1 - Завдання

**Хід роботи**

На початку було завантажено Git-репозиторій і отримано завдання за варіантом. Далі в \Lab12 було заповнено README.md файл і створено теки prj, Software, TestSuite та Report.

**Концептуалізація**

За умови задачі дано об’єкт – металева труба. За умовою вхідні дані – радіус і висота, вихідні – площа поперечного перерізу, тому абстрагуємо об’єкт до циліндра.

Обмеження щодо вхідних данних: 0.5 <= радіус <= 21.3, 0.5 <= висота <= 1200, значення беремо в сантиметрах. Дані взято з простору інтернету.

**Вимоги до модуля**

Властивості об’єкта – радіус, висота, оголошуються на рівні private. Методи отримання значень радіуса і висоти, методи ініціалізації полів класу та метод розрахунку площі поперечного перерізу оголошуються на рівні public.

При визначенні радіуса та висоти функції мають коректно зчитувати як десяткову крапку так і кому, якщо ж аргумент значення не приведено до форми стандартного типу float, то значення радіуса або висоти дорівнює нулю.

**Артефакти**

Приватні поля класу: height - висота цилідндра; radius - радіус циліндра;

Публічні методи класу: getHeight() – повертає значення приватної змінної height; getRadius() – повертає значення приватної змінної radius; setHeight() – функція приймає один аргумент – рядок, в якому всі коми замінюються на крапки, за допомогою бібліотеки <sstream>, виконується перевірка отриманого рядка на відповідність числовому значенню типу float. У випадку істинності виразу, приватне поле height ініціалізується даним рядком, приведеним до типу float методом stof(), інакше - ініціалізується значенням за замовчуванням – 0; setRadius(): ідентично до setHeight(), однак ініціалізується приватне поле radius; LateralSurfaceArea() – функція, в якості аргументів отримує висоту та радіус циліндра та повертає значення площі бокової поверхні.

**Тестові артефакти**

Вхідні значення та очікувані результати записані в текстовому файлі й приведені до спільної структури, що надалі буде використовуватись додатком Teacher.exe.

Вміст файлу TestSuite.txt:

*Висота: 1200*

*Радіус: 21.3*

*Площа поперечного перерізу: 160516.794250*

*Висота: 0.5*

*Радіус: 0.5*

*Площа поперечного перерізу: 1.570000*

*Висота: 0.4*

*Радіус: 5*

*Площа поперечного перерізу: 0.000000*

*Висота: 1201*

*Радіус: 5*

*Площа поперечного перерізу: 0.000000*

*Висота: 500*

*Радіус: 0.4*

*Площа поперечного перерізу: 0.000000*

*Висота: 500*

*Радіус: 22*

*Площа поперечного перерізу: 0.000000*

*Висота: text*

*Радіус: text*

*Площа поперечного перерізу: 0.000000*

*Висота: text*

*Радіус: 1.5*

*Площа поперечного перерізу: 0.000000*

*Висота: 225*

*Радіус: text*

*Площа поперечного перерізу: 0.000000*

*Висота: 23,4*

*Радіус: 1,5*

*Площа поперечного перерізу: 220.427996*

**Аналіз задачі 12.2**

Спочатку виконується перевірка, якщо файл main.cpp не знайдено в теці ../prj, то видається 100 звукових сигналів і в текстовий файл TestResults.txt записує рядок “Встановлені вимоги порядку виконання лабораторної роботи порушено!”, інакше проводиться unit-тестування класу з задачі 12.1, а саме:

Створюється об’єкт класу ClassLab12\_Tkachenko, файл TestSuite.txt відкривається для читання, а TestResults.txt – для запису. Відбувається перевірка на відкриття даних текстових файлів, якщо це не вдалося робота застосунку припиняться.

Потім за допомогою цикла на 10 ітерацій (кількість тест кейсів) виконується тестування: оголошення рядків, призначених для збереження радіусу, висоти та площі циліндра; зчитування з TestSuite.txt в рядок line методом getline(), починаючи з певного символа в рядку за допомогою erase(), відбувається для висоти, радіуса та площі; радіус, висота та площа ініціалізуються публічними методами, а площа приводиться до типу string та записується в змінну line; виконується виведення значень та отриманого результату Тест Кейсів у файл TestResults.txt.

**Висновки**

Під час підготовки до виконання лабораторної роботи було належно опрацьовано рекомендовану літературу та контрольні запитання. Після ознайомлення з порядком проведення лабораторної роботи, було розпочато її виконання.

Згідно завдання, було завантажено Git-репозиторій і в \Lab12 було створено теки prj, Software, TestSuite та Report.

Потім було виконано аналіз умови і постановку задач 12.1. У \Lab12\TestSuite\ було створено текстовий файл TestSuite.txt та збережено у ньому значення для тестування об’єкта класу ClassLab12\_Tkachenko. Потім в Code::Blocks IDE було відкрито заголовковий файл ModulesTkachenko з \Lab8\prj і за отриманими під час проектування програмного модуля артефактами виконано конструювання ADT ‒‒ класу С++, об’єкт якого за наданим інтерфейсом реалізовує розв’язування задачі 12.1. Так як проблем не виникло то я перейшов до завдання 2.

Було виконано аналіз і постановку задачі завдання 2.

Далі у Code::Blocks було створено в теці \prj проект консольного додатка, з назвою Teacher і виконано конструювання програмного засобу: мовою програмування С++ реалізувано проектні артефакти завдання 2. Потім Teacher.ехет було скопійовано його у \Lab12\Software\.

За допомогою розробленого додатка Teacher.ехе з \Software було виконано Unit-тестування об’єкта класу ClassLab12 \_ Tkachenko. Потім вихідний код заголовкового файлу ModulesПрізвище.h, проекта Teacher та вміст файлу TestResults.txt включено до звіту. Проблем не виникло тому написавши висновки я завершив виконання лабораторної роботи.

Процес виконання лабораторної роботи був повністю зрозумілим.

**Додаток А – вміст TestResults.txt**

*TEST CASE 1*

*Висота труби: 1200 cм*

*Радіус труби: 21.3 cм*

*Розрахована площа труби: 160516.794250 cм^2*

*Очікувана площа труби: 160516.794250 cм^2*

*TEST CASE RESULT: passed*

*TEST CASE 2*

*Висота труби: 0.5 cм*

*Радіус труби: 0.5 cм*

*Розрахована площа труби: 1.570000 cм^2*

*Очікувана площа труби: 1.570000 cм^2*

*TEST CASE RESULT: passed*

*TEST CASE 3*

*Висота труби: 0 cм*

*Радіус труби: 5 cм*

*Розрахована площа труби: 0.000000 cм^2*

*Очікувана площа труби: 0.000000 cм^2*

*TEST CASE RESULT: passed*

*TEST CASE 4*

*Висота труби: 0 cм*

*Радіус труби: 5 cм*

*Розрахована площа труби: 0.000000 cм^2*

*Очікувана площа труби: 0.000000 cм^2*

*TEST CASE RESULT: passed*

*TEST CASE 5*

*Висота труби: 500 cм*

*Радіус труби: 0 cм*

*Розрахована площа труби: 0.000000 cм^2*

*Очікувана площа труби: 0.000000 cм^2*

*TEST CASE RESULT: passed*

*TEST CASE 6*

*Висота труби: 500 cм*

*Радіус труби: 0 cм*

*Розрахована площа труби: 0.000000 cм^2*

*Очікувана площа труби: 0.000000 cм^2*

*TEST CASE RESULT: passed*

*TEST CASE 7*

*Висота труби: 0 cм*

*Радіус труби: 0 cм*

*Розрахована площа труби: 0.000000 cм^2*

*Очікувана площа труби: 0.000000 cм^2*

*TEST CASE RESULT: passed*

*TEST CASE 8*

*Висота труби: 0 cм*

*Радіус труби: 1.5 cм*

*Розрахована площа труби: 0.000000 cм^2*

*Очікувана площа труби: 0.000000 cм^2*

*TEST CASE RESULT: passed*

*TEST CASE 9*

*Висота труби: 225 cм*

*Радіус труби: 0 cм*

*Розрахована площа труби: 0.000000 cм^2*

*Очікувана площа труби: 0.000000 cм^2*

*TEST CASE RESULT: passed*

*TEST CASE 10*

*Висота труби: 23.4 cм*

*Радіус труби: 1.5 cм*

*Розрахована площа труби: 220.427996 cм^2*

*Очікувана площа труби: 220.427996 cм^2*

*TEST CASE RESULT: passed*

**Додаток Б – лістинг Teacher**

#include <iostream>

#include <fstream>

#include <Windows.h>

#include "ModulesTkachenko.h"

using namespace std;

bool fileInDirectory()

{

string cpp = \_\_FILE\_\_;

size\_t found = cpp.find("\\lab12\\prj");

if (found == string::npos) {

ofstream resFile("../TestSuite/TestResults.txt");

resFile << "Встановлені вимоги порядку виконання лабораторної роботи порушено!" << endl;

for (int i = 0; i < 100; i++)

cout << "\a";

resFile.close();

return false;

}

return true;

}

void TestSuiteResults(ofstream &results, ClassLab12\_Tkachenko pipe, string out\_area, string expected\_area, int index)

{

results << "TEST CASE " << index << endl;

results << "Висота труби: \t" << pipe.getHeight() << " cм" << endl;

results << "Радіус труби: \t" << pipe.getRadius() << " cм" << endl;

results << "Розрахована площа труби: \t" << out\_area << " cм^2" << endl;

results << "Очікувана площа труби: \t" << expected\_area << " cм^2" << endl;

if (out\_area == expected\_area)

results << "TEST CASE RESULT: passed\n"<< endl;

else

results << "TEST CASE RESULT: failed\n"<< endl;

}

int main()

{

SetConsoleOutputCP(1251);

if (fileInDirectory() == false) { return 0; }

ClassLab12\_Tkachenko pipe;

ifstream cases("../TestSuite/TestSuite.txt");

ofstream results("../TestSuite/TestResults.txt");

if (!cases.is\_open() || !results.is\_open())

{

cout << "Error: не вдалося відкрити файли." << endl;

return 0;

}

for (int i = 1; i < 11; i++) {

string pipe\_height;

string pipe\_radius;

string pipe\_area;

string line;

getline(cases, line);

pipe\_height = line.erase(0, 8);

getline(cases, line);

pipe\_radius = line.erase(0, 8);

getline(cases, line);

pipe\_area = line.erase(0, 28);

pipe.setHeight(pipe\_height);

pipe.setRadius(pipe\_radius);

line = to\_string(pipe.LateralSurfaceArea());

TestSuiteResults(results, pipe, line, pipe\_area, i);

}

cases.close();

results.close();

return 0;

}

**Додаток В – лістинг ModulesTkachenko.h**

#ifndef M\_H\_INCLUDED

#define M\_H\_INCLUDED

#define PI 3.14

#include <iostream>

#include <sstream>

#include <regex>

float s\_calculation(float x, float y, float z);

std::string function\_9\_1 (int ball\_bofort);

std::string function\_9\_2 (int girth\_of\_head);

std::string function\_9\_3 (int N);

void function\_10\_1();

void function\_10\_2();

void function\_10\_3(float x, float y, float z, int b);

using namespace std;

class ClassLab12\_Tkachenko

{

private:

double height, radius;

public:

void setHeight(string value);

double getHeight() { return height; }

void setRadius(string value);

double getRadius() { return radius; }

double LateralSurfaceArea() { return 2 \* PI \* radius \* height; }

};

void ClassLab12\_Tkachenko::setHeight(string value)

{

float f;

value = regex\_replace(value, regex(","), ".");

stringstream ss(value);

if (ss >> f)

{

if (stof(value) >= 0.5 && stof(value) <= 1200) //висота береться в см

height = stof(value);

else

height = 0;

}

else

height = 0;

}

void ClassLab12\_Tkachenko::setRadius(string value)

{

float f;

value = regex\_replace(value, regex(","), ".");

stringstream ss(value);

if (ss >> f)

{

if (stof(value) >= 0.5 && stof(value) <= 21.3) //радіус береться в см

radius = stof(value);

else

radius = 0;

}

else

radius = 0;

}

#endif // M\_H\_INCLUDED