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”

на тему

“ Програмна реалізація абстрактних типів диних”

**Виконав**

студент академічної

групи КБ-21

\_\_\_\_\_\_\_\_\_  Е.В. Ісаченков

**Приймає**

викладач кафедри

кібербезпеки та ПЗ

\_\_\_\_\_\_\_\_\_  П.С. Усік

Кропивницький – 2022

**Лабораторна робота №12**

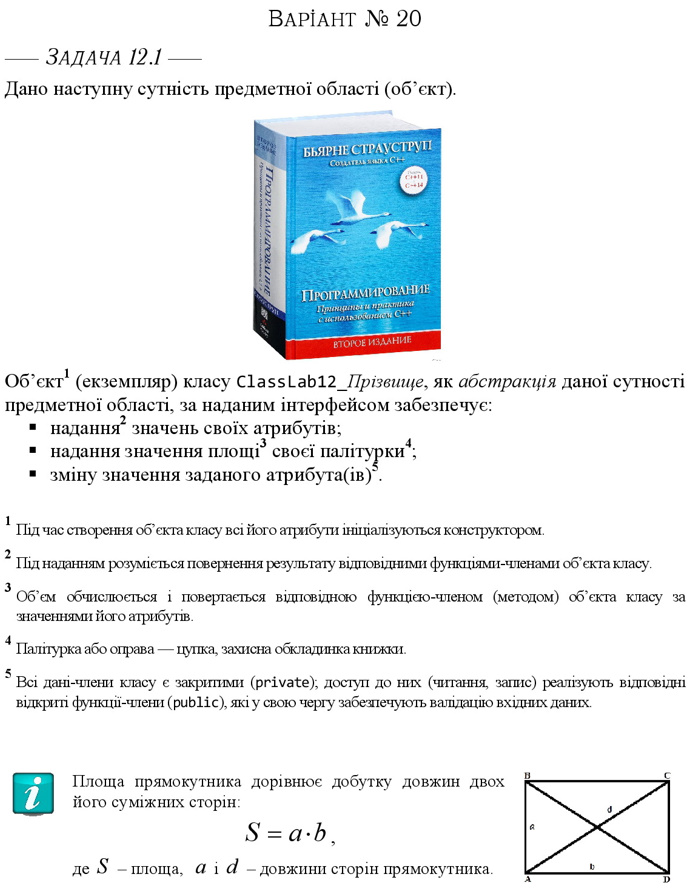
*Мета роботи* полягає у набутті ґрунтовних вмінь і практичних

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.

****

***Аналіз виконання Task 12.1:***

Мені потрібно зробити клас, як абстракція книжки. Для створення абстракції мені потрібно дві сторони палітурки, назвемо їх a і b. У мене буде два рівня доступу, це публічний та приватний. У приватному будуть знаходитися наші сторони a і b, в публічному будуть знаходитись функції. Клас має містити 5 функцій. Перша функція має виводити сторону a. Друга функція має теж виводити, але вже сторону b. Третя функція, повинна знаходити площу палітурки за допомогою сторін (a\*b) та виводити площу. Четверта функція повинна змінювати сторону a. П’ята функція повинна змінювати сторону b.

***Аналіз виконання Task 12.2:***

Потрібно реалізувати додаток Teacher, який буде видавати 100 звукових сигналів та записувати в файл «Встановлені вимоги порядку виконання лабораторної роботи порушено!», якщо файл main.срр під час його компіляції знаходився не в \Lab12\prj, інакше я роблю unit-тестування мого створеного класу. В тестування я буду задавати вхідні значення в файлі «TestSuite», буду зчитувати та завантажувати в клас дані. З цих даних в функції out\_S, буде знаходитися площа палітурки та співставлятимиться з очікуваним результатом. Який вказаний теж в файлі «TestSuite». Результат тестування буде записуватися в файл «TestResult».

***Висновок:***

*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. Для виконання лабораторної роботи №12 потрібно 2 академічні години. Потрібно таке обладнення:

1. Персональний комп’ютер з ОС Windows XP / Vista / 7 / 8.x / 10, Linux 32-bit / 64-bit або Mac OS X;
2. Вільне кросплатформове середовище розроблення програмного забезпечення Code::Blocks (www.codeblocks.org)
3. Текстовий редактор (OpenOffice Writer, Microsoft Word або ін.);
4. Git-репозиторій https://github.com/odorenskyi/Isachenkov-Eduard-KB-21;

Я дотримувався такого плану роботи:

Спочатку я завантажив свій Git-репозиторій та переглянув нове завдання.

Створив папку lab12 в ній теки prj, Report, Software, TestSuite, tasks та файл README.md.

Потім в текі Report створив Документ Word перейменував його, як «Звіт до Лабораторної роботи № 12», у ньому записав мету, номер варіанту, завдання, складу команди та титульну сторінку. Також в текстовий файл README, записав тему, мету, варіант та завдання.

Перейшов до аналізу завдання 12.1, відкривши методичні вказівки до лабораторної роботи №12.

Відкрив Code Blocks у ньому відкрив заголовковий файл ModulesIsachenkov.h та реалізував клас абстракції книжки.

Перейшов до аналізу задачі 12.2. На основі класу мені потрібно реалізувати модульне тестування.

Відкрив Code Blocks у ньому створив консольний додаток Teacher, мовою С++, створив програмний застосунок.

Створив в текі TestSuite два файла під назвою: «TestSuite» та «TestResult». В файлі «TestSuite» записав тест кейси для перевірки класу.

Провів тестування в результаті якого отримав в файлі «TestResult» позитивний статус тестування.

Відкрив «Звіт до Лабораторної роботи № 12», в ньому створив додатки в яких лістинги та вміст файлів, які були використані в лабораторній роботі №12.

Завантажив всі файли на Git-репозиторій.

Отже, закінчивши роботу я зрозумів що таке клас та навіщо його робити . За допомогою класу можна описувати різні не живі об’єкти, задавати їм характеристики та функції. Наприклад годинник, його функція крути стрілку для того щоб вказати точний час. За допомогою класу цей не живий об’єкт можна описати програмно. Ця робота мені сподобалась. Нажаль це остання лабораторна робота з предмету БМТП. За ці 12 лабораторних, я дізнався про таку цікаву мову програмування, як С++. Дякую, що навчили користуватися нею.

***Додаток №1:***

***Вихідний код модуля Teacher:***

#include <iostream>

#include <fstream>

#include <string>

#include "ModulesIsachenkov.h"

using namespace std;

int main()

{

string path = \_\_FILE\_\_;

string path\_result = "..\\TestSuite\\TestResult.txt";

string path\_suite = "..\\TestSuite\\TestSuite.txt";

size\_t check = path.find("\\lab12\\prj\\");

int expected\_a[5] = {1, 2, 3, 4, 5};

ClassLab12\_Isachenkov Test;

if(check == string::npos){

ofstream fout;

fout.open(path\_result);

fout << "Встановлені вимоги порядку виконання лабораторної роботи порушено!";

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

cout << "\a";

fout.close();

}

else {

ifstream fin;

ofstream fout;

string a, b, text;

fout.open(path\_result);

fin.open(path\_suite);

for(int i = 0; i < 5; i++){

getline(fin, a, ',');

getline(fin, b, ';');

getline(fin, text);

double s = stoi(text);

Test.in\_a(stoi(a));

Test.in\_b(stoi(b));

if(Test.out\_S() == s){

fout << "Test case " << i+1 << " PASSED\n";

}

else

fout << "Test case " << i+1 << " FAILED\n";

}

fout.close();

fin.close();

}

system("pause");

return 0;

}

***Вихідний код заголовкового файлу ModulesIsachenkov.h:***

#ifndef MODULESISACHENKOV\_H\_INCLUDED

#define MODULESISACHENKOV\_H\_INCLUDED

class ClassLab12\_Isachenkov {

public:

ClassLab12\_Isachenkov () { a = 5.0, b = 5.0;}

ClassLab12\_Isachenkov (float side\_a, float side\_b) {a = side\_a, b = side\_b;}

float out\_a () {return a;}

float out\_b () {return b;}

double out\_S () {return a\*b;}

void in\_a (float side\_a) {a = side\_a;}

void in\_b (float side\_b) {b = side\_b;}

private:

float a, b;

};

#endif // MODULESISACHENKOV\_H\_INCLUDED

***Додаток №2:***

***TestSuite(вхідний текст файлу TestSuite):***

2,5;10

3,3;9

4,1;4

5,5;25

4,7;28

***Виведення результату тестування в файл(вихідний текст файлу TestResult):***

Test case 1 PASSED

Test case 2 PASSED

Test case 3 PASSED

Test case 4 PASSED

Test case 5 PASSED