

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

ЗВІТ
ПРО ВИКОНАННЯ ЛАБОРАТОРНОЇ РОБОТИ № 8
з навчальної дисципліни
“Базові методології та технології програмування”
РЕАЛІЗАЦІЯ СТАТИЧНИХ БІБЛІОТЕК МОДУЛІВ ЛІНІЙНИХ
ОБЧИСЛЮВАЛЬНИХ ПРОЦЕСІВ

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

ВИКОНАВ
студент академічної групи КІ-22-2
Дощенко Ю.Д.

ПЕРЕВІРИВ
ст. викладач кафедри кібербезпеки
та програмного забезпечення
Собінов О.Г.

Лабораторна робота №8

Тема: реалізація статичних бібліотек модулів лінійних обчислювальних процесів

Мета роботи: полягає у набутті ґрунтовних вмінь і практичних навичок застосування теоретичних положень методології модульного програмування, реалізації метода функціональної декомпозиції задач, метода модульного (блочного) тестування, представлення мовою програмування C++ даних скалярних типів, арифметичних і логічних операцій, потокового введення й виведення інформації, розроблення програмних модулів та засобів у кросплатформовому середовищі Code::Blocks (GNU GCC Compiler).

Завдання:

1. Реалізувати статичну бібліотеку модулів libModulesПрізвище C/C++, яка містить функцію розв'язування задачі 8.1.
2. Реалізувати програмне забезпечення розв'язування задачі 8.2 — консольний застосунок.

Варіант 12

1.

<https://github.com/odorenskyi/>

ВАРІАНТ 12

— ЗАДАЧА 8.1 —

За значеннями x, y, z обчислюється S :

$$S = \frac{\sin x}{\sqrt{\frac{yz}{|x+y|}}} + 3y^5.$$

Малюнок 1 – Умова задачі 8.1

Строга постановка задачі:

Вхідні дані: x, y, z – дійсні числа:

Вихідні дані: S – дійсне число.

Проектування програмного модуля:

Модуль ModulesDoshchenko складається із заголовкового файлу, що містить оголошення прототипу функцій, та файлу вихідного коду, в якому зберігається реалізація функції `s_calculation`.

Функція `s_calculation` приймає три аргументи в якості вхідних даних та повертає результат виразу, зазначеного в умові.

Тестовий драйвер TestDriver – функція, призначена для тестування функції. Вона містить:

- масив з вхідними даними;
- масив з очікуваними результатами;
- цикл для перебору наборів значень та виводу результату тестування певного набору.

Тест-сьют модульного тестування статичної бібліотеки ModulesDoshchenko.a наведений у додатку А.

Лістинги вихідного коду проєктів ModulesDishschenko та TestDriver містяться в Додатку Б та Додатку В відповідно.

Результати тестування s_calculation зі статичної бібліотеки libModulesDoshchenko.a тестовим драйвером:

```

TestDriver of ModulesDoshchenko.a
TC_01: |x =1  y=2  z =3      | - result: - passed(96.4207)
TC_02: |x =4  y=5  z =6      | - result: - passed(9374.75)
TC_03: |x =7  y=8  z =9      | - result: - passed(98304.2)
TC_04: |x =7  y=8  z =2      | - result: - passed(98304.7)
TC_05: |x =2  y=1  z =8      | - result: - passed(3.45465)

```

2.

— ЗАДАЧА 8.2 —

За послідовними запитам вводяться числа x, y, z та символи a і b .

Вивести (включити у потік STL — cout)*:

8.2.1. Прізвище та ім'я розробника програми зі знаком охорони авторського права «©» (від англ. copyright);

8.2.2. Результат логічного виразу в числовому вигляді (1/0):

$$a + 1 < b + 3 ?$$

8.2.3. Значення x, y, z в десятковій і шістнадцятковій системах числення; S , що обчислюється функцією s_calculation() заголовкового файлу ModulesПрізвище.h.

Малюнок 2 – Умова задачі 8.2

Строга постановка задачі:

Вхідні дані: x, y, z – цілі числа,

a, b – символьні літери;

Вихідні дані: прізвище та ім'я розробника з символом «©»,
результат логічного виразу,

значення x , z в десятковій та шістнадцятковій системах числення,
 S , що обчислюється функцією `s_calculation()` заголовкового файлу `ModulesDoshchenko.h` ;

Висновок: дана лабораторна робота була націлена на набуття навичок у використанні теоретичних положень модульної парадигми програмування, реалізації метода функціональної декомпозиції задач, метода модульного тестування, а також у представленні мовою програмування C++ даних скалярних типів, арифметичних і логічних операцій, потокового введення та виведення інформації, розробленні програмних модулів та засобів у кросплатформовому середовищі `Code::Blocks` (GNU GCC Compiler). Також важливою частиною лабораторної роботи було набуття навичок з використання системи контролю версій файлів та спільної роботи `Git`.

На відміну від процедурної парадигми, де всі методи реалізовані в одному файлі вихідного коду, що викликає незручності в достатньо масштабних проектах, модульна парадигма базується на понятті «модуль», яке можна пояснити як самостійну програмну одиницю, що служить для виконання певної функції програми та для зв'язку з іншою частиною програми. В даній лабораторній роботі створений модуль було представлено у вигляді статичної бібліотеки (файл з розширенням `.a`), яка була підключена до проектів за допомогою лінкера та до файлу вихідного коду за допомогою новоствореного заголовкового файлу (файл з розширенням `.h`), що містить прототипи функцій, що містяться в статичній бібліотеці.

Статична бібліотека була створена шляхом відповідного проекту в застосунку `Code::Blocks`. Далі для подальшого інтегрування бібліотеки у проекти, написані мовою C++, файл `main.c` було замінено на `main.cpp`, після чого потребувалося додати новостворений файл у проект. У даному файлі було реалізовано функцію для вирахування виразу, заданого в умові завдання, що повертала значення результату. Після компіляції проекту статичної бібліотеки в теці `\obj` було створено файл з розширенням `.a`, що і представляє з себе файл статичної бібліотеки.

Наступним кроком було створення заголовкового файлу, який містив прототип функції, реалізованої в статичній бібліотеці.

Далі було створено проект консольного додатка C++ під назвою `TestDriver`. Метою створення цього додатка була реалізація концепції модульного тестування. Модульне тестування, відоме як Unit-тестування, використовується для автоматизованого тестування модулів шляхом порівняння значення, що повертається функцією з модуля, яка приймає набір еталонних вхідних даних (аргументів), з еталонним результатом для відповідного набору. Важливою ідеєю юніт-тестування є така: якщо хоча б один з тестів провалився, після відлагодження модуля потрібно проводити повторне проведення тестів, що йшли перед провальним тестом, до моменту, коли усі тести проходять успішно.

Після цього було створено ПЗ з реалізацією функцій з наступного завдання. Важливою умовою було проектування функцій з інтерфейсом, який

підходить для виводу у вихідний потік (`cout`). Для коректного виводу знаку © довелось намагатись писати через команди `string`, `cout`, `authorCopyright()` та просто вставляти знак копірайту але через недостані навички та можливу помилку в програмі мені не вдалось зробити знак копірайту тож я вирішив просто зробити так (C).

Важливою частиною лабораторної роботи було використання системи контролю версій файлів та спільних роботи Git. Ідея системи контролю версій полягає у зручному зберіганні різних версій проекту в одному місці. Замість зберігання копій одного й того ж проекту, реєструються зміни в проекті, що дозволяє за потреби відкатити проект до потрібної версії (в даному контексті “версія” описує стан проект після певної послідовності змін). У випадку цієї лабораторної роботи проект зберігається на сайті Github. Для відсилання змін у проекті на сайт подрібно за допомогою застосунку `git-scm` клонувати репозиторій проекту на свій пристрій, зробити потрібні зміни. Далі додати потрібні файли в список для додання до комміту (структурна одиниця в архітектурі `git`), після чого потрібно сформувати комміт з коментарем до нього. Останнім кроком є пуш(відсилання) комміту на сервер. Для сайту Github у застосунку `git-scm` потрібно попередньо авторизувати свій акаунт, який повинен мати доступ для змін до потрібного репозиторію. Історія змін складається з коммітів, кожен з яких має свій ідентифікатор, назву (коментар), дату додання та посилання на акаунт, від імені якого відбувся пуш. Система `git` дозволяє дізнаватись про те, які саме зміни були внесені, ким і коли, що значно спрощує командну розробку проектів.

ДОДАТОК А

Назва тестового набору Test Suite Description	TS_lab8_1
Назва проекту / Модуля Name of Project / Unit	ModulesDoshchenko.a
Рівень тестування Level of Testing	модульний / Unit Testing
Автор тест-сююта Test Suite Author	Дощенко Юрій
Виконавець Implementer	Дощенко Юрій

Ід-р тест- кейса / Test Case ID	Вхідні дані / Input values	Очікуваний результат / Expected Result	Результат тестування / Test Result
TC_01	x = 1 y= 2 z = 3	96.4207	96.4207
TC_02	x = 4 y=5 z = 6	9374.75	9374.75
TC_03	x = 7 y=8 z = 9	98304.2	98304.2
TC_04	x = 7 y=8 z = 2	98304.7	98304.7
TC_05	x = 2 y=1 z = 8	3.45465	3.45465

ДОДАТОК Б

Лістинг вихідного коду проекту ModulesDoshchenko:

```
#include <iostream>
#include <cmath>

float s_calculation(int x, int y, int z){
    float result = sin(x) / abs((y*z)/(x+y)) + 3 * pow(y, 5);
    return result;
}
```

ДОДАТОК Б

Лістинг вихідного коду проекту TestDriver:

```
#include <iostream>
#include "ModulesDoshchenko.h"
#include <Windows.h>

using namespace std;

string print_name(){
    return "Юрій Дощенко ©";
}

bool logic(char a,char b){
    return a+1<b+3;
}

float rebut(int x, int y, int z){

    cout << "x = " << dec << x << " У шістнадцятковій системі: " << hex << x << endl;
    cout << "y = " << dec << y << " У шістнадцятковій системі: " << hex << y << endl;
    cout << "z = " << dec << z << " У шістнадцятковій системі: " << hex << z << endl;
    return s_calculation(x,y,z);
}

int main()
{
    SetConsoleCP(1251);
    SetConsoleOutputCP(1251);
    int x, y, z;
    char a,b;
    cout << print_name() << endl;
    cin >> a >> b;
    cout << logic(a,b) << endl;
    cin >> x >> y >> z;
    cout << rebut(x,y,z) << endl;
    return 0;
}
```


ДОДАТОК Г

Назва тестового набору Test Suite Description	TS_lab8_2
Назва проекту / ПЗ Name of Project / Software	Doshchenko_task.exe
Рівень тестування Level of Testing	системний / System Testing
Автор тест-сюїта Test Suite Author	Дощенко Юрій
Виконавець Implementer	Дощенко Юрій

Ід-р тест- кейса / Test Case ID	Дії (кроки) / Action (Test Steps)	Очікуваний результат / Expected Result	Результат тестування / Test Result
TC_01	<ol style="list-style-type: none"> Ввести "0 0" Натиснути Enter Ввести "1 2 3" Натиснути Enter 	<p>Дощенко Юрій (с) Академ-група:KI-22-2 ЦНТУ Введіть значення a та b 0 0 Результат:1</p> <p>Введіть значення x,y та z 1 2 3 В десятичній системі числення x,y та z відповідно: 1 2 3 В шістнадцятковій системі числення x,y та z відповідно: 1 2 3 Результат виразу у функції s_calculation():96 . 4207</p>	passed
TC_02	<ol style="list-style-type: none"> Ввести "1 2" Натиснути Enter Ввести "4 5 6" Натиснути Enter 	<p>Дощенко Юрій (с) Академ-група:KI-22-2 ЦНТУ Введіть значення a та b 1 2 Результат:1 Введіть значення x,y та z В десятичній системі числення x,y та z відповідно: 4 5 6 В шістнадцятковій системі числення x,y та z відповідно: 4 5 6 Результат виразу у функції s_calculation():9374 . 75</p>	passed
TC_03	<ol style="list-style-type: none"> Ввести "3 4" Натиснути Enter Ввести "7 8 9" Натиснути Enter 	<p>Дощенко Юрій (с) Академ-група:KI-22-2 ЦНТУ Введіть значення a та b 3 4 Результат:1 Введіть значення x,y та z В десятичній системі числення x,y та z відповідно: 7 8 9 В шістнадцятковій системі числення x,y та z відповідно: 7 8 9 Результат виразу у функції s_calculation():98304 . 2</p>	passed

TC_04	<ol style="list-style-type: none"> 1. Ввести "5 6" 2. Натиснути Enter 3. Ввести "7 8 2" 4. Натиснути Enter 	<p>Дощенко Юрій (с) Академ-група:KI-22-2 ЦНТУ</p> <p>Введіть значення a та b</p> <p>5 6 Результат:1</p> <p>Введіть значення x,y та z</p> <p>В десятковій системі числення x,y та z відповідно:</p> <p>7 8 2</p> <p>В шістнадцятковій системі числення x,y та z відповідно:</p> <p>7 8 2</p> <p>Результат виразу у функції</p> <p>s_calculation():98304 . 7</p>	passed
TC_05	<ol style="list-style-type: none"> 1. Ввести "7 8" 2. Натиснути Enter 3. Ввести "2 1 8" 4. Натиснути Enter 	<p>Дощенко Юрій (с) Академ-група:KI-22-2 ЦНТУ</p> <p>Введіть значення a та b</p> <p>7 8 Результат:1</p> <p>Введіть значення x,y та z</p> <p>В десятковій системі числення x,y та z відповідно:</p> <p>2 1 8</p> <p>В шістнадцятковій системі числення x,y та z відповідно:</p> <p>2 1 8</p> <p>Результат виразу у функції</p> <p>s_calculation():3 . 45465</p>	passed