Міністерство освіти і науки України

Центральноукраїнський національний технічний університет

Механіко-технологічний факультет

Кафедра кібербезпеки та програмного забезпечення

# ЗВІТ

Про виконання лабораторної роботи № 10

з навчальної дисципліни “Базові методології та технології програмування”

РЕАЛІЗАЦІЯ ПРОГРАМНИХ МОДУЛІВ ОБРОБЛЕННЯ ДАНИХ СКЛАДОВИХ ТИПІВ З ФАЙЛОВИМ ВВЕДЕННЯМ/ВИВЕДЕННЯМ

ВИКОНАВ

студент академічної групи

КІ-21-1 Василенко Д.М.

Перевірив асистент

Усік П.С \_\_\_\_\_\_\_\_\_\_\_

Кропивницький – 2022

**Мета роботи:** полягає у набутті ґрунтовних вмінь і практичних навичок реалізації у Code::Blocks IDE мовою програмування С++ програмних модулів створення й оброблення даних типів масив, структура, об’єднання, множина, перелік, перетворення типів даних, використання файлових потоків та функцій стандартних бібліотек для оброблення символьної інформації.

**Завдання до лабораторної роботи:**

1. Реалізувати програмні модулі розв’язування задач 10.1–10.3

як складові статичної бібліотеки libModulesПрізвище.а (проект

ModulesПрізвище лабораторних робіт №8–9).

1. Реалізувати тестовий драйвер автоматизованої перевірки

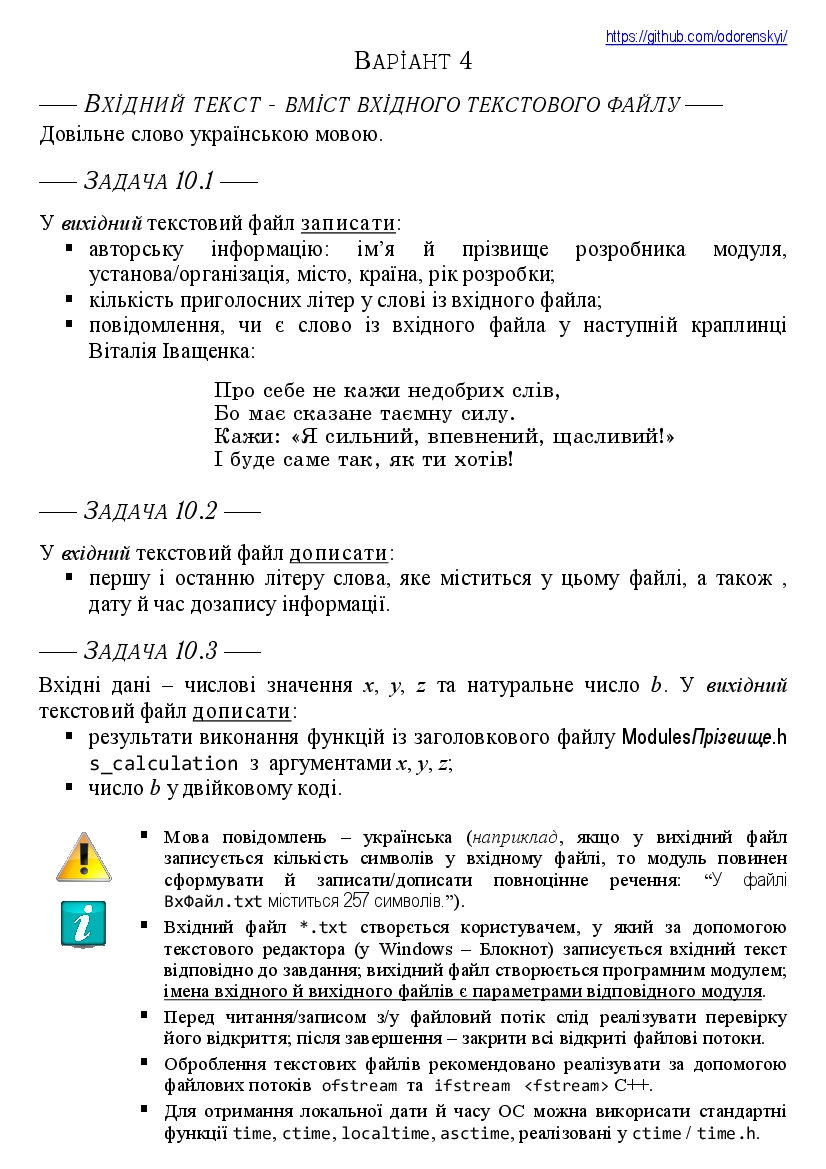
програмних модулів розв’язування задач 10.1–10.3.

Рисунок 1.1 – Завдання варіанту №4

**Формалізація задачі 10.1**

**Вихідні дані:**

* авторська іформація;
* *amount* – кількість приголосних літер у слові з *InFile;*

amount ∈ [0; 2147483648];

* повідомлення про наявність слова з *InFile* у краплинці.

**Вхідні дані:**

*OutFile* – ім'я вихідного файлу;

*InFile* - ім'я вхідного файлу.

**Формалізація задачі 10.2**

**Вихідні дані:**

* slovo та lastChar – перша і остання літери слова, яке міститься у вхідному файлі
* дата й час дозапису інформації.

**Вхідні дані:**

*InOutFile* - ім'я вхідного та вихідного файлу.

**Формалізація задачі 10.3**

**Вихідні дані:**

* результат виконання функції s\_calculation;
* число b у двійковому коді.

**Вхідні дані:**

*OutFile* – ім'я вихідного файлу;

**Результат тестування:**

Тестування функцiї до задачi 10.1

Test case #1 is passed!

Test case #2 is passed!

Тестування функцiї до задачi 10.2

Test case #1 is passed!

Test case #2 is passed!

Тестування функцiї до задачi 10.3

Test case #1 is passed!

Test case #2 is passed!

**Висновок:**

У ході виконання даної лабораторної роботи були реалізовані функції розв’язування задач 10.1 – 10.3 як складових статичної бібліотеки libModulesVasylenko.a (проект ModulesVasylenko, створений під час виконання лабораторної роботи №8 та доповнений під час виконання лабораторної роботи №9) та був реалізований тестовий драйвер для автоматизованої перевірки програмних модулів розв'язування задач 10.1 – 10.3 зі статичної бібліотеки libModulesVasylenko.a

По-перше, почергово було виконано аналіз вимог, постановка задач, проектування архітектури та детальне проектування програмних модулів розв’язування задач 10.1 – 10.3. Після цього розроблено три набори контрольних прикладів до задач 10.1 – 10.3 для виконання автоматизованого модульного тестування одержаних функцій (див. Додатки А, Б, В).

Потім була виконана реалізація функцій мовою програмування С++ за отриманими під час проектування програмних модулів артефактами. Проект статичної бібліотеки ModulesVasylenko, який вже містить реалізовану функцію s\_calculation та функції, створені під час виконання лабораторної роботи №9, доповнено функціями, які за наданим інтерфейсом реалізують розв’язування задач 10.1, 10.2 та 10.3 (див. Додаток Г). Проект заголовкового файлу ModulesVasylenko доповнено прототипами (заголовками) функцій розв’язування задач 10.1 -10.3.

Реалізовано консольний застосунок - тестовий драйвер для виконання автоматизованого модульного тестування функцій розв’язування задач 10.1, 10.2 та 10.3 за допомогою розроблених наборів контрольних прикладів та вхідних та вихідних тестових файлів.

За допомогою розробленого тестового драйвера виконано автоматизоване модульне тестування функцій розв’язування задач 10.1 – 10.3 з бібліотеки libModulesVasylenko.a. (див. Додаток Д) Результат виконання всіх тест-кейсів – passed, що означає, що модуль працює згідно поставленим умовам. Результати модульного тестування функцій 10.1 – 10.3 статичної бібліотеки libModulesVasylenko.a задокументовані (див. Рис.2)

В результаті виконання лабораторної роботи мною були набутті ґрунтовні вміння і практичні навички реалізації у Code::Blocks IDE мовою програмування С++ програмних модулів створення й оброблення даних типів масив, структура, об'єднання, множина, перелік, перетворення типів даних, використання файлових потоків та функцій стандартних бібліотек для оброблення символьної інформації.

**Додаток А**

|  |  |
| --- | --- |
| Назва тестового набору  Test Suite Description | TS\_10\_1 |
| Назва проекта / ПЗ  Name of Project / Software | ModulesVasylenko |
| Рівень тестування  Level of Testing | модульний / Unit Testing |
| Автор тест-сьюта  Test Suite Author | Dima Vasylenko |
| Виконавець  Implementer | Dima Vasylenko |

|  |  |  |  |
| --- | --- | --- | --- |
| Вхідні дані /  Preliminary Steps | Дії (кроки) /  Action (Test Steps) | Очікуваний  результат /  Expected Result | Результат тестування /  Test Result |
| Файл: InPut1.txt  Вміст файлу: "Ого" | Виклик модуля "task\_10\_1" з вхідними даними "InPut1.txt" та "OutPut1.txt" | Вміст файлу OutPut1.txt:  Розробник: Василенко Д.М.  м. Кропивницький, Україна  ЦНТУ, 2022  Кількість приголосних літер у слові "Ого": 1  Cлово "Ого" відсутнє в краплинці Віталія Іващенка. | passed |
| Файл: InPut2.txt  Вміст файлу: "слово" | Виклик модуля "task\_10\_1" з вхідними даними "InPut2.txt" та "OutPut2.txt" | Вміст файлу OutPut2.txt:  Розробник: Василенко Д.М.  м. Кропивницький, Україна  ЦНТУ, 2022  Кількість приголосних літер у слові "слово": 3  Cлово "слово" відсутнє в краплинці Віталія Іващенка. | passed |

**Додаток Б**

|  |  |
| --- | --- |
| Назва тестового набору  Test Suite Description | TS\_10\_2 |
| Назва проекта / ПЗ  Name of Project / Software | ModulesVasylenko |
| Рівень тестування  Level of Testing | модульний / Unit Testing |
| Автор тест-сьюта  Test Suite Author | Dima Vasylenko |
| Виконавець  Implementer | Dima Vasylenko |

|  |  |  |  |
| --- | --- | --- | --- |
| Вхідні дані /  Preliminary Steps | Дії (кроки) /  Action (Test Steps) | Очікуваний  результат /  Expected Result | Результат тестування /  Test Result |
| Файл: InPut1.txt  Вміст файлу: "Ого" | Виклик модуля "task\_10\_2" з вхідними даними "InPut1.txt" | Вміст файлу InPut1.txt:  Ого  Перша і остання літери слова: [О] [о] | passed |
| Файл: InPut2.txt  Вміст файлу: "слово" | Виклик модуля "task\_10\_2" з вхідними даними "InPut2.txt" | Вміст файлу InPut2.txt:  слово  Перша і остання літери слова: [с] [о] | passed |

**Додаток В**

|  |  |
| --- | --- |
| Назва тестового набору  Test Suite Description | TS\_10\_3 |
| Назва проекта / ПЗ  Name of Project / Software | ModulesVasylenko |
| Рівень тестування  Level of Testing | модульний / Unit Testing |
| Автор тест-сьюта  Test Suite Author | Dima Vasylenko |
| Виконавець  Implementer | Dima Vasylenko |

|  |  |  |  |
| --- | --- | --- | --- |
| Вхідні дані /  Preliminary Steps | Дії (кроки) /  Action (Test Steps) | Очікуваний  результат /  Expected Result | Результат тестування /  Test Result |
| Вхідні дані: х=2, z=2, b=3  Та файл OutPut1.txt | Виклик модуля "task\_10\_3" з вхідними даними "OutPut1.txt" | Розробник: Василенко Д.М.  м. Кропивницький, Україна  ЦНТУ, 2022  Кількість приголосних літер у слові "Ого": 1  Cлово "Ого" відсутнє в краплинці Віталія Іващенка.  Результат виконання функції s\_calculation: S = 155.674  Число 3 у двійковому коді: 00000000000000000000000000000011 | passed |
| Вхідні дані: х=5, z=3, b=2  Та файл OutPut2.txt | Виклик модуля "task\_10\_3" з вхідними даними "OutPut2.txt" | Вміст файлу OutPut2.txt:  Розробник: Василенко Д.М.  м. Кропивницький, Україна  ЦНТУ, 2022  Кількість приголосних літер у слові "Слово": 3  Cлово "Слово" відсутнє в краплинці Віталія Іващенка.  Результат виконання функції s\_calculation: S = 79.6486  Число 2 у двійковому коді: 00000000000000000000000000000010 | passed |

**Додаток Г**

#include "ModulesVasylenko.h"

#include <cmath>

#include <fstream>

#include <string>

#include <ctime>

#include <bitset>

#include <vector>

#include <math.h>

#include <cmath>

#ifndef M\_PI

#define M\_PI (3.14159265358979323846)

#endif

using namespace std;

double s\_calculation(double x, double y, double z)

{

double S;

S = z + M\_PI\*((pow((2\*z + 1), 2) - sqrt(abs(y - 0.5\*z))/sqrt(cos(z + y\*z)+ pow(x, 2))));

return S;

}

float Temperature9\_2(float t)

{

return (5\*(t-32))/9;

}

int Binary9\_3(int N)

{

int result = 0;

bool flag = false;

for (int n=sizeof(int)\*8-1; n>-1; n--)

{

if((N>>n)&1)

if(!flag)

flag = true;

if (flag)

{

if((N>>4)&1)

result += ((N>>n)&1)? 1 : 0;

else

result += ((N>>n)&1)? 0 : 1;

}

}

return result;

}

void task\_10\_1(char \*OutFile,char \*InFile)

{

ofstream outFile;

outFile.open(OutFile);

outFile << "Розробник: Василенко Д.М.\n"

"м. Кропивницький, Україна\n"

"ЦНТУ, 2022\n";

ifstream inFile;

inFile.open(InFile);

string fromFile;

inFile >> fromFile;

inFile.close();

int amount = 0;

for (int i=0; i<fromFile.length(); i++)

{

if (!(fromFile[i] == 'А' || fromFile[i] == 'а' || fromFile[i] == 'Е' || fromFile[i] == 'е' || fromFile[i] == 'Є' || fromFile[i] == 'є' || fromFile[i] == 'І' || fromFile[i] == 'і' || fromFile[i] == 'Ї' || fromFile[i] == 'ї' || fromFile[i] == 'О' || fromFile[i] == 'о' || fromFile[i] == 'У' || fromFile[i] == 'у' || fromFile[i] == 'Ю' || fromFile[i] == 'ю' || fromFile[i] == 'И' || fromFile[i] == 'и'))

amount++;

}

outFile << "\nКількість приголосних літер у слові \"" << fromFile << "\": " << amount << endl;

string kraplina = "Про себе не кажи недобрих слів,"

"Бо має сказане таємну силу."

"Кажи:""Я сильний, впевнений, щасливий!"

"І буде сам так, як ти хотів!";

if(kraplina.find(fromFile) != string::npos)

outFile << "Cлово \"" << fromFile << "\" є в краплинці Віталія Іващенка." << endl;

else

outFile << "Cлово \"" << fromFile << "\" відсутнє в краплинці Віталія Іващенка." << endl;

outFile.close();

}

void task\_10\_2(char \*OutInFile, string slovo)

{

ifstream inFile;

inFile.open(OutInFile);

string fromFile;

inFile >> fromFile;

inFile.close();

ofstream in\_File;

in\_File.open(OutInFile, ios::app);

char lastChar;

lastChar = slovo.back();

in\_File << "\n Перша і остання літери слова: " << "[" << slovo[0] << "]" << " [" << lastChar << "]" << endl;

time\_t times = time(NULL);

in\_File << "Час дозапису :" << asctime (localtime( &times )) << endl;

in\_File.close();

}

void task\_10\_3(float x, float z, float b, char \*OutFile)

{

ofstream outFile;

outFile.open(OutFile, ios::app);

outFile << "\nРезультат виконання функції s\_calculation: S = " << s\_calculation(x,z,b) << endl;

if (b>0)

outFile<< "Число " << b << " у двійковому коді: " << bitset<32>(b) << endl;

else

outFile<< "Число " << b << " не натуральне." << endl;

outFile.close();

}

**Додаток Д**

#include <iostream>

#include "ModulesVasylenko.h"

#include <fstream>

#include <string>

#include <ctime>

void task\_10\_2(char \*OutInFile, string slovo)

{

ifstream inFile;

inFile.open(OutInFile);

string fromFile;

inFile >> fromFile;

inFile.close();

ofstream in\_File;

in\_File.open(OutInFile, ios::app);

int amoun = 0;

char lastChar;

lastChar = slovo.back();

in\_File << "\n Перша і остання літери слова: " << "[" << slovo[0] << "]" << " [" << lastChar << "]" << endl;

time\_t times = time(NULL);

in\_File << "Час дозапису :" << asctime (localtime( &times )) << endl;

in\_File.close();

}

using namespace std;

int main()

{

char \*locale=setlocale(LC\_ALL, "");

//Тестування функції до задачі 10.1

cout << "Тестування функцiї до задачi 10.1\n";

string test\_cases\_out[2][6] ={

{{"Розробник: Василенко Д.М."},

{"м. Кропивницький, Україна"},

{"ЦНТУ, 2022"},

{""},

{"Кількість приголосних літер у слові \"Ого\": 1"},

{"Cлово \"Ого\" відсутнє в краплинці Віталія Іващенка."}},

{{"Розробник: Василенко Д.М."},

{"м. Кропивницький, Україна"},

{"ЦНТУ, 2022"},

{""},

{"Кількість приголосних літер у слові \"слово\": 3"},

{"Cлово \"слово\" відсутнє в краплинці Віталія Іващенка."}}

};

string test\_cases\_in[2] = {{"Ого"},

{"слово"}};

// Test case#1

ofstream create\_file("InPut1.txt");

create\_file << test\_cases\_in[0];

create\_file.close();

char OutFile1[] = "OutPut1.txt";

char InFile1[] = "InPut1.txt";

task\_10\_1(OutFile1,InFile1);

ifstream read\_from\_file;

read\_from\_file.open("OutPut1.txt");

string for\_compare;

int fail = 0;

for (int i = 0; i<6; i++)

{

getline(read\_from\_file, for\_compare);

if (test\_cases\_out[0][i] == for\_compare)

continue;

else

fail++;

}

read\_from\_file.close();

if(fail==0)

cout << "Test case #1 is passed!\n";

else

cout << "Test case #1 is failed...\n";

// Test case#2

fail = 0;

ofstream create\_file2("InPut2.txt");

create\_file2 << test\_cases\_in[1];

create\_file2.close();

char OutFile2[] = "OutPut2.txt";

char InFile2[] = "InPut2.txt";

task\_10\_1(OutFile2,InFile2);

ifstream read\_from\_file2;

read\_from\_file2.open("OutPut2.txt");

string for\_compare2;

for (int i = 0; i<6; i++)

{

getline(read\_from\_file2, for\_compare2);

if (test\_cases\_out[1][i] == for\_compare2)

continue;

else

fail++;

}

read\_from\_file2.close();

if(fail==0)

cout << "Test case #2 is passed!\n";

else

cout << "Test case #2 is failed...\n";

//Тестування функції до задчі 10.2

cout << "Тестування функцiї до задачi 10.2\n";

string test\_cases2[2][2] = {{{"Ого"},

{" Перша і остання літери слова: [О] [о]"},

},

{{"слово"},

{" Перша і остання літери слова: [с] [о]"},

}};

// Test case#1

string from\_file2;

task\_10\_2(InFile1,"Ого");

ifstream read\_file;

read\_file.open("InPut1.txt");

fail = 0;

for (int i = 0; i<2; i++)

{

getline(read\_file, from\_file2);

if (test\_cases2[0][i] == from\_file2)

continue;

else

fail++;

}

read\_file.close();

if(fail==0)

cout << "Test case #1 is passed!\n";

else

cout << "Test case #1 is failed...\n";

// Test case#2

string from\_file2\_2;

task\_10\_2(InFile2,"слово");

ifstream read\_file2;

read\_file2.open("InPut2.txt");

fail = 0;

for (int i = 0; i<2; i++)

{

getline(read\_file2, from\_file2\_2);

if (test\_cases2[1][i] == from\_file2\_2)

continue;

else

fail++;

}

read\_file2.close();

if(fail==0)

cout << "Test case #2 is passed!\n";

else

cout << "Test case #2 is failed...\n";

//Тестування функції до задчі 10.3

cout << "Тестування функцiї до задачi 10.3\n";

string test\_cases3[2][9] = {{{"Розробник: Василенко Д.М."},

{"м. Кропивницький, Україна"},

{"ЦНТУ, 2022"},

{""},

{"Кількість приголосних літер у слові \"Ого\": 1"},

{"Cлово \"Ого\" відсутнє в краплинці Віталія Іващенка."},

{""},

{"Результат виконання функції s\_calculation: S = 155.674"},

{"Число 3 у двійковому коді: 00000000000000000000000000000011"},

},

{{"Розробник: Василенко Д.М."},

{"м. Кропивницький, Україна"},

{"ЦНТУ, 2022"},

{""},

{"Кількість приголосних літер у слові \"слово\": 3"},

{"Cлово \"слово\" відсутнє в краплинці Віталія Іващенка."},

{""},

{"Результат виконання функції s\_calculation: S = 79.6486"},

{"Число 2 у двійковому коді: 00000000000000000000000000000010"},

}};

// Test case#1

string from\_file3\_1;

task\_10\_3(2, 2, 3, OutFile1);

ifstream read\_file3;

read\_file3.open("OutPut1.txt");

fail = 0;

for (int i = 0; i<9; i++)

{

getline(read\_file3, from\_file3\_1);

if (test\_cases3[0][i] == from\_file3\_1)

continue;

else

fail++;

}

read\_file3.close();

if(fail==0)

cout << "Test case #1 is passed!\n";

else

cout << "Test case #1 is failed...\n";

// Test case#2

string from\_file3\_2;

task\_10\_3(5, 3, 2, OutFile2);

ifstream read\_file4;

read\_file4.open("OutPut2.txt");

fail = 0;

for (int i = 0; i<9; i++)

{

getline(read\_file4, from\_file3\_2);

if (test\_cases3[1][i] == from\_file3\_2)

continue;

else

fail++;

}

read\_file4.close();

if(fail==0)

cout << "Test case #2 is passed!\n";

else

cout << "Test case #2 is failed...\n";

cin.get();

return 0;

}