Міністерство освіти і науки України

Центральноукраїнський національний технічний університет

Механіко-технологічний факультет

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

ЗАВДАННЯ ВИДАВ

доцент кафедри кібербезпеки та програмного забезпечення

Доренський О. П.

[https://github.com/odorenskyi/](https://github.com/odorenskyi/Dmytro-Parkhomenko-KB18)

ВИКОНАВ

студент академічної групи

КБ-21

Полях О.В

ПЕРЕВІРИВ

викладач кафедри

кібербезпеки

та програмного забезпечення

Усік П.С

Кропивницький – 2022

# ЛАБОРАТОРНА РОБОТА №10

**Тема:** Реалізація програмних модулів оброблення даних складових типів з файловим введенням/виведенням.

**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.

## Варіант 18

**Завдання**

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

як складові статичної бібліотеки libModulesПрізвище.а (проект

ModulesПрізвище лабораторних робіт №8–9).

2.Реалізувати тестовий драйвер автоматизованої перевірки

програмних модулів розв’язування задач 10.1–10.3.

**Хід роботи**

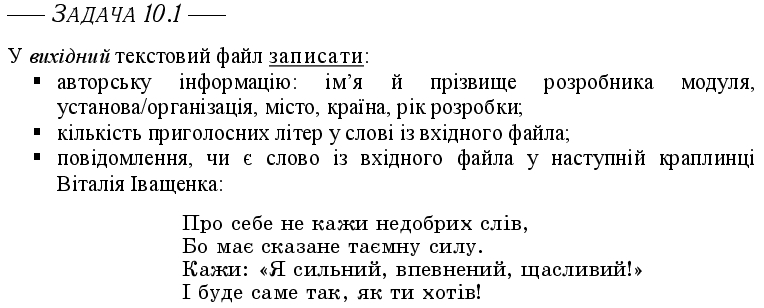


Рисунок 1.1 - Умова задачі 10.1

Вхідні дані: слово в форматі string записане до вхідного файлу

Вихідні дані: записані до вихідного файлу: авторська інформація, кількість приголосних літер, пошук слова в краплинці.

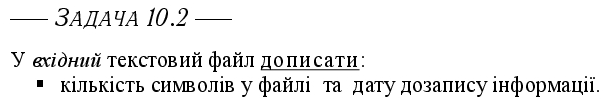


Рисунок 1.2 - Умова задачі 10.2

Вхідні дані: слово записане до вхідного файлу

Вихідні дані: записані до вихідного файлу: кількість символів у файлі, дата та час дозапису.

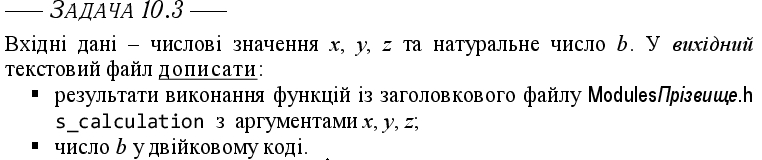


Рисунок 1.3 - Умова задачі 10.3

Вихідні та вхідні дані вказані в умові задачі.

**Результати виконання TestDriver**

10.1

Слово: true

Запис авторської інформації: true

Кількість прииголосних літер у слові:true

Повідомлення,чи є слово: true

Дозапис часу редагування файлу: true

Результату функції s\_calculation: true

Конвертування десяткового числа в двійкове: true

10.2

Слово: true

Запис авторської інформації: true

Кількість прииголосних літер у слові:true

Повідомлення,чи є слово: true

Дозапис часу редагування файлу: true

Результату функції s\_calculation: true

Конвертування десяткового числа в двійкове: true

10.3

Слово: true

Запис авторської інформації: true

Кількість прииголосних літер у слові:true

Повідомлення,чи є слово: true

Дозапис часу редагування файлу: true

Результату функції s\_calculation: true

Конвертування десяткового числа в двійкове: true

**Висновок**:

Мета лабораторної роботи № 10 з дисципліни “Базові методології та технології програмування на тему “Реалізація програмних модулів оброблення даних складових типів з файловим введенням/виведенням” полягає у набутті ґрунтовних вмінь і практичних навичок реалізації у Code::Blocks IDE мовою програмування C++ програмних модулів створення й оброблення даних типів масив, структур, об'єднання, множина, перелік, перетворення типів даних, використання файлових потоків та функцій стандартних бібліотек для оброблення символьної інформації.

Лабораторна робота виконувалась згідно «Порядок виконання лабораторної роботи та методичні вказівки». Лабораторна робота № 10 стала другою лабораторною роботою з використанням системи контроля версій – GitHub. Як вже було зазначено в моєму попередньому звіті з лабораторної роботи №10 з дисципліни “Базові методології та технології програмування на тему “Реалізація статичних бібліотек модулів лінійних обчислювальних процесів” мета якої 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 С++ 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) GitHub дає можливість розбити роботу на етапи, яка є дуже зручною, особливо сьогодні.

Для вирішення завдань лабораторної роботи був скопійований ModulesPoliakh з 8 та 9 лабораторної роботи,реалізована статистична бібліотека libModulesPoliakh.a,а в середині знаходяться функції s\_calculcation,authorcopyright,prov\_world,Timestamp, binaryInDecimal,calculation та підключені бібліотеки:сmath,clocale,cstring,string,iostream,bitset,fstream,ctime,stdio.h. Також був реалізований заголовковий файл ModulesPoliakh.h в середині якого знаходиться прототипи функцій. Для того, щоб підключити створену бібліотеку було також налаштовано опції проєкта Build options…:Linker; Compiler.Перевірив всі результати в TestDriver.Для кожної задачі був розроблений TestSuite як в лабораторних роботах.

Лабораторна робота №10 стала першою і складнішою роботою.Потрібно було працювати с текстовими файлами,спочатку я довго шукав інформацію про текстові файли,знайшов інформацію про бібліотеку fstream,для вирішення задач я використовув бібліотеку fstream,за допомогою якої я працював с текстовими файлами.

Отже, під час виконання складної як для мене лабораторної роботи № 10 з дисципліни “Базові методології та технології програмування на тему “Реалізація статичних бібліотек модулів лінійних обчислювальних процесів” я набув ґрунтовних вмінь і практичних навичок реалізації у Code::Blocks IDE мовою програмування C++ програмних модулів створення й оброблення даних типів масив, структур, об'єднання, множина, перелік, перетворення типів даних, використання файлових потоків та функцій стандартних бібліотек для оброблення символьної інформації.

**Додаток А**

(Тестові набори)

|  |  |
| --- | --- |
| Назва тестового набору  Test Suite Description | TS\_10 |
| Назва проекта / ПЗ  Name of Project / Software | ModulesPoliakh |
| Рівень тестування  Level of Testing | модульний / Unit testing |
| Автор тест-сьюта  Test Suite Author | Полях Олександр |
| Виконавець  Implementer | Полях Олександр |

|  |  |  |  |
| --- | --- | --- | --- |
| Ід-р тест-кейса / Test Case ID | Дії (кроки) /  Action  (Test Steps) | Очікуваний  результат /  Expected Result | Результат тестування (пройшов/не вдалося/ заблокований) /  Test Result (passed/failed/ blocked) |
| 10.1 | 1.Довільне слово: сказане  2.Записати авторську інформацію  3.Кількість приголосних літер у слові:  4.Повідомлення,чи є слово із вхідного файла  5.Виконати функцію s\_calculation (x = 9, y = 8, z = 7)  6.Конвертувати десяткове число 123 в двійкове | Вікно застосунку:  Слово: true  Запис авторської інформації: true  Кількість прииголосних літер у слові:true  Повідомлення,чи є слово: true  Дозапис часу редагування файлу: true  Результату функції s\_calculation: true  Конвертування десяткового числа в двійкове: true  Вхідний файл:  сказане  Дата та час дозапису інформаціїї: Fri Jun 03 22:43:37 2022  Вихідний файл:  -----------------------------------------------  Розробник: Полях Олександр  ВНЗ: Центральний Національний Технічний Університет  Рік розробки: 2022  Місто/Країна: Кропивницький/Україна  ---------------------------------------------------  Слово Є!  Кількість приголосних літер: 3  Результат виконання функції s\_calculation: --182850  Число 123 в двійковій системі числень: 00000000000000000000000001111011 | passed |
| 10.2 | 1.Довільне слово: 12334  2.Записати авторську інформацію  3. Кількість приголосних літер у слові  4. Повідомлення,чи є слово із вхідного файла 5.Виконати функцію s\_calculation (x = 6, y = 5, z = 4)  6.Конвертувати 100 десяткове число в двійкове | Вікно застосунку:  Довільне слово: true  Запис авторської інформації: true  Повідомлення,чи є слово із вхідного файла:false  Дозапис часу редагування файлу: true  Результату функції s\_calculation: true  Конвертування десяткового числа в двійкове: true  Вхідний файл:  12334  Дата та час дозапису інформаціїї: Fri Jun 03 22:43:37 2022  Вихідний файл:  -----------------------------------------------  Розробник: Полях Олександр  ВНЗ: Центральний Національний Технічний Університет  Рік розробки: 2022  Місто/Країна: Кропивницький/Україна  ---------------------------------------------------  Cлова немає  Результат виконання функції s\_calculation: -7049.32  Число 100 в двійковій системі:  00000000000000000000000001100100 | passed |
| 10.3 | 1.Записати рядок: кіт  2.Записати авторську інформацію  3.Порахувати кількість символів верхнього регістра  4.Зробити пошук введеного рядка  5.Виконати функцію s\_calculation (x = 3, y = 2, z = 1)  6.Конвертувати десяткове число 52 в двійкове | Вікно застосунку:  Слово: true  Запис авторської інформації: true  Кількість прииголосних літер у слові:true  Повідомлення,чи є слово: false  Дозапис часу редагування файлу: true  Результату функції s\_calculation: true  Конвертування десяткового числа в двійкове: true  Вхідний файл:  кіт  Дата та час дозапису інформаціїї: Fri Jun 03 22:43:37 2022  Вихідний файл:  -----------------------------------------------  Розробник: Полях Олександр  ВНЗ: Центральний Національний Технічний Університет  Рік розробки: 2022  Місто/Країна: Кропивницький/Україна  ---------------------------------------------------  Слово немає  Кількість приголосних літер: 1  Результат виконання функції s\_calculation:  -2.51252  Число 5 в двійковій системі числень: 00000000000000000000000000000101 | passed |

**Додаток Б**

(Лістинг ModulesPoliakh)

#include <cmath>

#include <clocale>

#include <cstring>

#include <string>

#include <iostream>

#include <bitset>

#include <fstream>

#include <ctime>

#include <stdio.h>

using namespace std;

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

return 2.72 \* abs(sin(z)) + pow(0.5,pow(x,2)) - sqrt(abs(pow(y + z,2) - pow(x,5))) / 10 \* 3.14 \* pow(z,4);

}

double waveHeight(double wHeight)

{

int grade = 0;

if(wHeight == 0){

return grade;

}

if(wHeight >= 0 && wHeight <= 0.1){

grade = 1;

return grade;

}

if(wHeight >= 0.1 && wHeight <= 0.5){

grade = 2;

return grade;

}

if(wHeight >= 0.5 && wHeight <= 1.25){

grade = 3;

return grade;

}

if(wHeight >= 1.25 && wHeight <= 2.50){

grade = 4;

return grade;

}

if(wHeight >= 2.50 && wHeight <= 4.0){

grade = 5;

return grade;

}

if(wHeight >= 4 && wHeight <= 6){

grade = 6;

return grade;

}

if(wHeight >= 6 && wHeight <= 9){

grade = 7;

return grade;

}

if(wHeight >= 9 && wHeight <= 14){

grade = 8;

return grade;

}

if(wHeight > 14){

grade = 9;

return grade;

}

}

int temperature\_celsia(float farengeit){

float celsia = 0.55555555556 \* (farengeit - 32);

return celsia;

}

int bits\_number(int number){

bitset<32> b\_number{number};

if(b\_number[1]){

return b\_number.count();

}

return 32 - b\_number.count();

}

void authorcopyright(){

string line;

fstream in("D:\\input.txt");

if(in.is\_open()){

while(getline(in,line)){

cout << line;

}

}

int abcd = 0;

for(int i = 0; i < line.size(); i++){

if(line[i] != 'а' && line[i] != 'е' && line[i] != 'у' && line[i] != 'і' && line[i] != 'ї' && line[i] != 'о' && line[i] != 'и' && line[i] != 'є' && line[i] != 'ю' && line[i] != 'я')

{

abcd++;

}

cout << line.size();

}

ofstream copyright;

copyright.open("D:\\output.txt");

if(copyright.is\_open()){

copyright << " ----------------------------------------------- " << endl

<< " Розробник: Полях Олександр " << endl

<< " ВНЗ: Центральний Національний Технічний Університет " << endl

<< " Рік розробки: 2022 " << endl

<< " Місто/Країна: Кропивницький/Україна " << endl

<< " --------------------------------------------------- " << endl << endl

<< "Кількість приголосних літер: " << abcd << endl;

}

copyright.close();

}

void prov\_world()

{

string line;

string krap\_1 = "Про себе не кажи недобрих слів";

string krap\_2 = "Бо має сказане таємну силу.";

string krap\_3 = "Кажи:Я сильний, впевнений, щасливий!";

string krap\_4 = "І буде сам так, як ти хотів!";

fstream in("D:\\input.txt");

if (in.is\_open())

{

while(getline(in, line))

{

cout << line << endl;

}

}

int tmp = 0;

for(int i = 0; i < 4; i++)

{

for(int j = 0; j < krap\_1.size() || j < krap\_2.size() ||j < krap\_3.size() ||j < krap\_4.size();j++)

{

if(krap\_1[j] == line[j] &&

krap\_1[j+1] == line[j+1] &&

krap\_1[j+2] == line[j+2] &&

krap\_1[j+3] == line[j+3])

{

tmp = 1;

ofstream text;

text.open("D:\\output.txt", ios::app);

if(text.is\_open())

{

text << "Слово Є!";

}

break;

}

if(krap\_2[j] == line[j] &&

krap\_2[j+1] == line[j+1] &&

krap\_2[j+2] == line[j+2] &&

krap\_2[j+3] == line[j+3])

{

tmp = 1;

ofstream text;

text.open("D:\\output.txt", ios::app);

if(text.is\_open())

{

text << "Слово Є!";

}

break;

}

if(krap\_3[j] == line[j] &&

krap\_3[j+1] == line[j+1] &&

krap\_3[j+2] == line[j+2] &&

krap\_3[j+3] == line[j+3])

{

tmp = 1;

ofstream text;

text.open("D:\\output.txt", ios::app);

if(text.is\_open())

{

text << "Слово Є!";

}

break;

}

if(krap\_4[j] == line[j] &&

krap\_4[j+1] == line[j+1] &&

krap\_4[j+2] == line[j+2] &&

krap\_4[j+3] == line[j+3])

{

tmp = 1;

ofstream text;

text.open("D:\\output.txt", ios::app);

if(text.is\_open())

{

text << "Слово Є!";

}

break;

}

}

}

if(tmp = 0){

ofstream text\_1;

text\_1.open("D:\\output.txt", ios::app);

if(text\_1.is\_open()){

text\_1 << "Слова немає!";

}

}

}

void Timestamp(){

ofstream timestamp;

timestamp.open("D:\\input.txt", ios::app);

time\_t rawtime;

time(&rawtime);

timestamp << "Дата та час дозапису інформаціїї: " << ctime(&rawtime) << endl;

timestamp.close();

}

void calculation(){

ofstream outputFile;

outputFile.open("D:\\output.txt", ios::app);

outputFile << "Результат виконання функції s\_calculation: " << s\_calculation << endl;

outputFile.close();

}

void binaryInDecimal(int b){

ofstream outputFile;

outputFile.open("D:\\output.txt", ios::app);

bitset<32> b\_number(b);

outputFile << "Число " << b << " в двійковій системі числень: " << bits\_number << endl;

outputFile.close();

}

**Додаток В**

(Лістинг TestDriver)

#include <iostream>

#include "ModulesPoliakh.h"

#include <cstring>

#include <conio.h>

#include <iomanip>

using namespace std;

int main()

{

system("chcp 1251 & cls");

string inputtext [3] = {"сказане",

"12334",

"кіт"};

double x [3] = {9,6,3};

double y [3] = {8,5,2};

double z [3] = {7,4,1};

int b [3] = {123,100,5};

double resultS [3] = {-182850, -7049.32, -2.51252};

double resultB [3] = {00000000000000000000000001111011,00000000000000000000000001100100,00000000000000000000000000000101};

for(int i = 0; i < 3; i++) {

cout << boolalpha

<< "Запис авторської інформації: " << authorcopyright << endl;

cout << boolalpha

<< "Кількість приголосних літер:"<< authorcopyright << endl;

cout << boolalpha

<< "Повідомлення,чи є слово у краплинці: " << prov\_world << endl;

cout << boolalpha

<< "Дозапис часу редагування файлу: "<< Timestamp << endl;

cout << boolalpha

<< "Результату функції s\_calculation: " << (s\_calculation(x[i], y[i], z[i]) <= resultS[i] + 0.005) << endl;

cout << boolalpha << "Конвертування десяткового числа в двійкове: " << bits\_number << endl;

getch();

}

return 0;

}

**Додаток Г**

(ModulesPoliakh.h)

#ifndef MODULESPOLIAKH\_H\_INCLUDED

#define MODULESPOLIAKH\_H\_INCLUDED

#include <cstring>

using namespace std;

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

double waveHeight(double wHeight);

int temperature\_celsia(float farengeit);

int bits\_number(int number);

void authorcopyright();

void prov\_world();

void Timestamp();

void calculation();

void binaryInDecimal(int b);

#endif // MODULESPOLIAKH\_H\_INCLUDED