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 КН-23

Горбенко К.О.

ПЕРЕВІРИВ

ст. викладач кафедри кібербезпеки   
та програмного забезпечення

Дрєєва Г. М.

Кропивницький – 2024

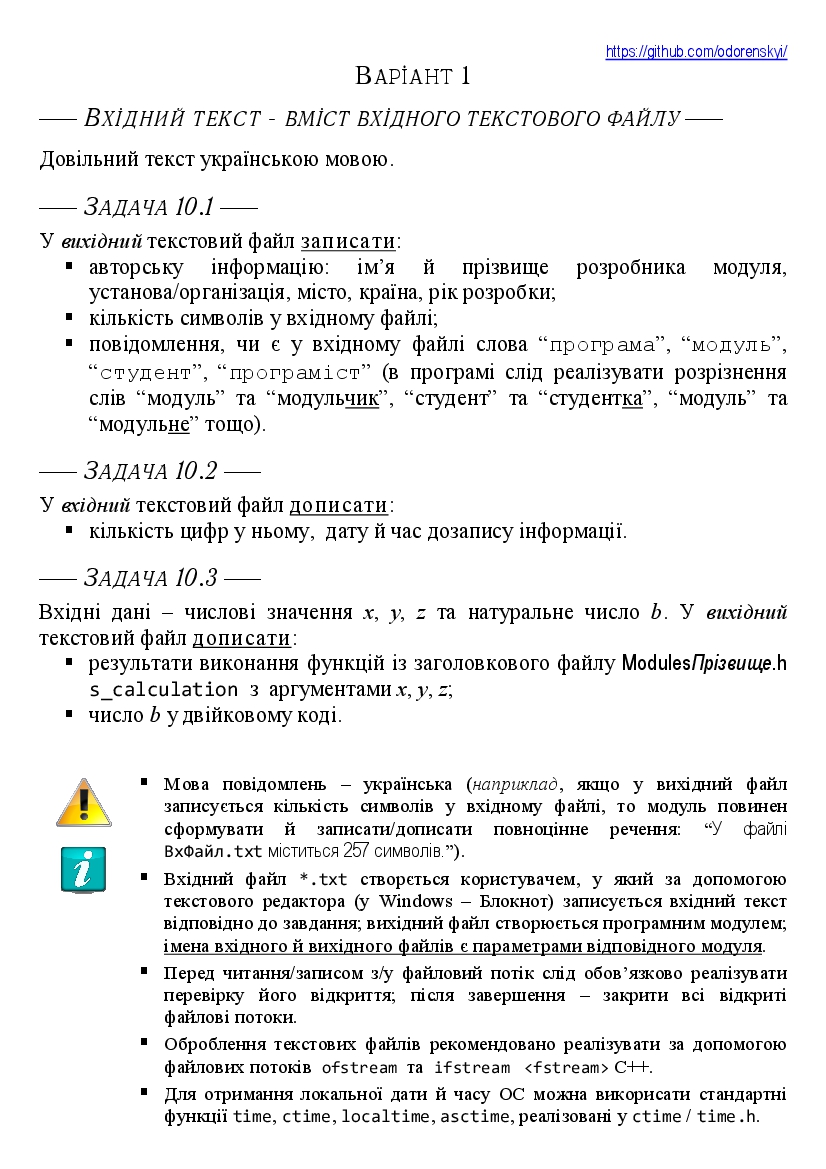
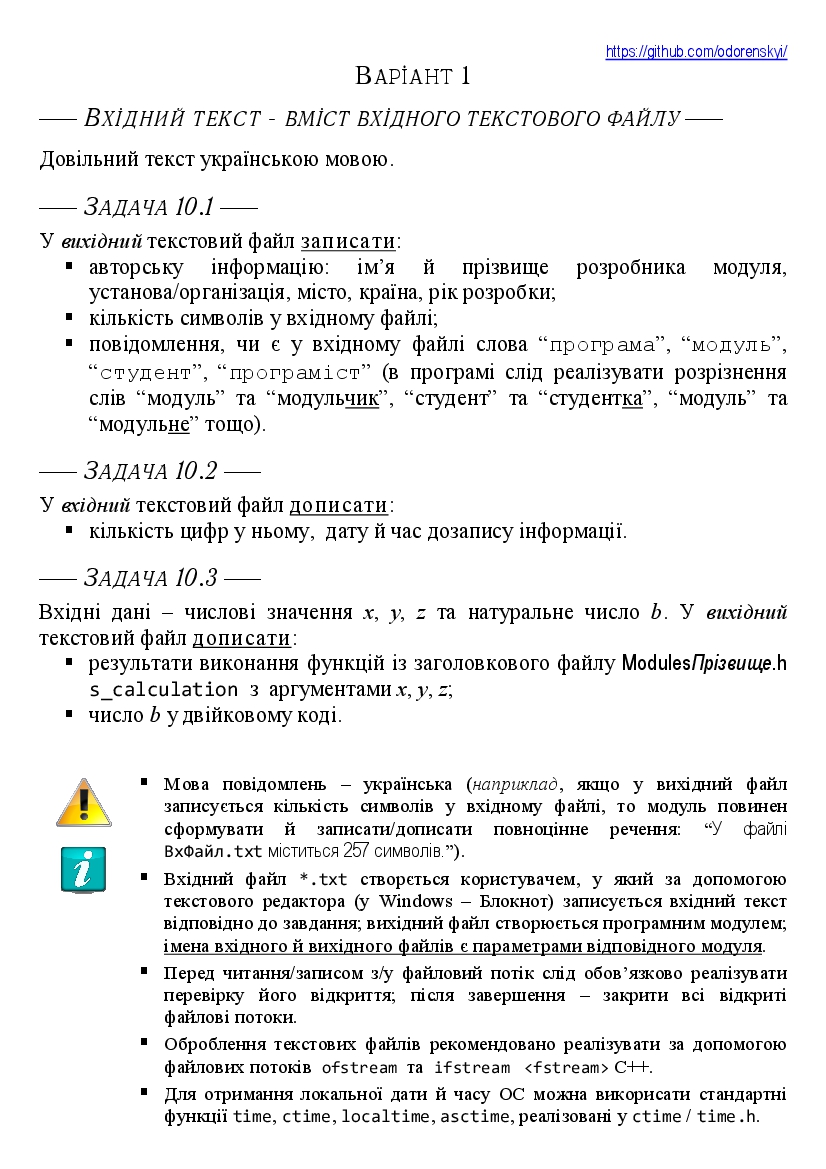
**Лабораторна робота №10**

*Тема***:** Реалізація програмних модулів розгалужених та ітераційних обчислювальних процесів

*Мета***:** Набуття 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).
2. Реалізувати тестовий драйвер автоматизованої перевірки програмних модулів розв’язування задач 10.1–10.3



***Завдання 10.1***

*Вхідні дані:* inputFileName, outputFileName – вхідний та вихідний фали

*Вихідні дані:* вихідний текстовий файл в якому міститься авторська інформація, кількість символів файлу inputFileName та повідомленя про вміст слів у файлі (“програма””,”модуль”,”студент”, “програміст”)

*Опис:* Щоб могти працювати з файлами нам потрібно використовувати файлові потоки <fstream>. Ми використовуємо inputFile для вхідного файлу і outputFile для вихідного файлу . Після читання вмісту файлу у зміну content, ми обчислюється кількість символів у рядку content за допомогою методу size(). Далі створюються масиви words і wordFound, які містять слова для пошуку та прапорці їх наявності. В циклі слова з масиву words шукаються у рядку content. Якщо слово знайдено, відповідний прапорець у масиві wordFound встановлюється в true.

***Завдання 10.2***

*Вхідні дані*: filename – вхідний файл

*Вихідні дані:* у вхідний файл filename виконується до запис інформації про кількість чисел та про час запису інформації

*Опис:* По прикладу першого завдання ми знову використовуємо ofstream щоб відкрити файл для до запису, та ifstream для зчитування. Потім ми отримуємо поточний час за допомогою функцій time. Після в нас йде цикл який зчитує файл посимвольно та перевіряє, чи є зчитаний символ цифрою якщо є то збільшує лічильник на один.

***Завдання 10.3***

*Вхідні дані:* x, y, z – числові значення типу float, b – натуральне число типу unsigned int, outputFileName – вихідний файл

*Вихідні дані:* Результат виконання функції з ModulesHorbenko s\_calculation з аргументами x, y, z. Число b у двійковому коді.

*Опис:* Подібно до минулих завдань здійснюємо відкриття файлу для до запису. З ModuleHorbenko використовуємо функцію s\_calculation та виводимо результат у файл разом з аргументами. Після в нас запуусскається цикл for який починається з значущого біта і веде цикл до менш значущого, в циклі використовується маска яка отримає біт на позиції i. В кінці циклу відбувається до запис за допомогою тернарного оператора ‘1’ або ‘0’.

***Реалізація виконання TestDriver:***

=============Test\_10.1==============

Test\_1

====================================

Операція завершена. Результати записані у вихідний файл.

Перевірити виконання того що тест-сьюта та результата програми,

якщо воно сходиться натисніть 'y', якщо ні то іншу кнопку

Passed

====================================

Test\_2

====================================

Операція завершена. Результати записані у вихідний файл.

Перевірити виконання того що тест-сьюта та результата програми,

якщо воно сходиться натисніть 'y', якщо ні то іншу кнопку

Passed

====================================

Test\_3

====================================

Операція завершена. Результати записані у вихідний файл.

Перевірити виконання того що тест-сьюта та результата програми,

якщо воно сходиться натисніть 'y', якщо ні то іншу кнопку

Passed

====================================

Test\_4

====================================

Операція завершена. Результати записані у вихідний файл.

Перевірити виконання того що тест-сьюта та результата програми,

якщо воно сходиться натисніть 'y', якщо ні то іншу кнопку

Passed

====================================

Test\_5

====================================

Операція завершена. Результати записані у вихідний файл.

Перевірити виконання того що тест-сьюта та результата програми,

якщо воно сходиться натисніть 'y', якщо ні то іншу кнопку

Passed

====================================

=============Test\_10.2==============

Test\_1

====================================

Перевірте зміст файлу з тест-сьютом,

якщо воно сходиться натисніть 'y', якщо ні то іншу кнопку

Passed

====================================

Test\_2

====================================

Перевірте зміст файлу з тест-сьютом,

якщо воно сходиться натисніть 'y', якщо ні то іншу кнопку

Passed

====================================

Test\_3

====================================

Перевірте зміст файлу з тест-сьютом,

якщо воно сходиться натисніть 'y', якщо ні то іншу кнопку

Passed

====================================

Test\_4

====================================

Перевірте зміст файлу з тест-сьютом,

якщо воно сходиться натисніть 'y', якщо ні то іншу кнопку

Passed

====================================

Test\_5

====================================

Перевірте зміст файлу з тест-сьютом,

якщо воно сходиться натисніть 'y', якщо ні то іншу кнопку

Passed

====================================

=============Test\_10.3==============

Test\_1

====================================

Перевірте зміст файлу з тест-сьютом,

якщо воно сходиться натисніть 'y', якщо ні то іншу кнопку

Passed

====================================

Test\_2

====================================

Перевірте зміст файлу з тест-сьютом,

якщо воно сходиться натисніть 'y', якщо ні то іншу кнопку

Passed

====================================

Test\_3

====================================

Перевірте зміст файлу з тест-сьютом,

якщо воно сходиться натисніть 'y', якщо ні то іншу кнопку

Passed

====================================

Test\_4

====================================

Перевірте зміст файлу з тест-сьютом,

якщо воно сходиться натисніть 'y', якщо ні то іншу кнопку

Passed

====================================

Test\_5

====================================

Перевірте зміст файлу з тест-сьютом,

якщо воно сходиться натисніть 'y', якщо ні то іншу кнопку

Passed

***Лістинг ModulesHorbenko***

#include <cmath>

#include <iostream>

#include <string>

#include <fstream>

#include <ctime>

using namespace std;

//=======================LAB8.1=======================

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

{

if (x < 2)

{

cout << "'Nan' Помилка, значення X менше 2 " << endl;

return 0;

}else

{

return pow(3 \* sin(sqrt(12 \* x + log10(x - 3))), y) + (z / x);

}

}

//=======================LAB9.1=======================

string tornado\_category(int speed)

{

string category;

if (speed >= 64 && speed <= 116) {

category = "Категорія торнадо: FO\nЧастота: 38,9%";

} else if (speed >= 117 && speed <= 180) {

category = "Категорія торнадо: F1\nЧастота: 35,6%";

} else if (speed >= 181 && speed <= 253) {

category = "Категорія торнадо: F2\nЧастота: 19,4%";

} else if (speed >= 254 && speed <= 332) {

category = "Категорія торнадо: F3\nЧастота: 4,9%";

} else if (speed >= 333 && speed <= 418) {

category = "Категорія торнадо: F4\nЧастота: 1,1%";

} else if (speed >= 419 && speed <= 512) {

category = "Категорія торнадо: F5\nЧастота: менше 0,1%";

} else {

category = "Веденно не коректне значення. Ведіть число від 64 до 512";

}

return category;

}

//=======================LAB9.2=======================

float\* calculate\_temp(float temp1, float temp2, float temp3, float temp4, float temp5, float temp6)

{

static float array[2];

float sum = temp1 + temp2 + temp3 + temp4 + temp5 + temp6;

float average\_celsius = sum / 6;

float average\_fahrenheit = (average\_celsius \* 9 / 5) + 32;

array[0] = average\_celsius;

array[1] = average\_fahrenheit;

return array;

}

//=======================LAB9.3=======================

unsigned int Bits(unsigned int num) {

int count = 0;

int first\_bit = 1;

while (first\_bit <= num) {

if ((num & first\_bit) == 0) {

count++;

}

first\_bit <<= 1;

}

return (num >> 15) ? (16 - count) : count;

}

//=======================LAB10.1=======================

void processfile(const string& inputFileName, const string& outputFileName) {

ifstream inputFile(inputFileName);

ofstream outputFile(outputFileName);

string content((istreambuf\_iterator<char>(inputFile)), istreambuf\_iterator<char>());

inputFile.close();

int charCount = content.size();

string words[] = {"програма", "модуль", "студент", "програміст"};

bool wordFound[] = {false, false, false, false};

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

size\_t pos = content.find(words[i]);

while (pos != string::npos) {

wordFound[i] = true;

pos = content.find(words[i], pos + 1);

}

}

outputFile << " ====================================" << endl;

outputFile << "||Автор: Горбенко Катерина ||" << endl;

outputFile << "||Установа: Центральноукраїнський ||" << endl;

outputFile << "||національний технічний університет||" << endl;

outputFile << "||Місто: Кропівницький ||" << endl;

outputFile << "||Країна: Україна ||" << endl;

outputFile << "||Рік розробки: 2024 ||" << endl;

outputFile << " ====================================" << endl;

outputFile << "Кількість символів у вхідному файлі: " << charCount << endl;

outputFile << "======================================" << endl;

outputFile << "Результати перевірки на наявність слів:" << endl;

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

outputFile << words[i] << ": " << (wordFound[i] ? "Знайдено" : "Не знайдено") << endl;

}

outputFile << "======================================" << endl;

outputFile.close();

cout << "Операція завершена. Результати записані у вихідний файл." << endl;

}

//=======================LAB10.2=======================

void appendfileInfo(const string& filename) {

ofstream file(filename, ios\_base::app);

time\_t currentTime = time(nullptr);

string dateTime = asctime(localtime(&currentTime));

ifstream inputFile(filename);

int digitCount = 0;

char ch;

while (inputFile.get(ch)) {

if (isdigit(ch)) {

digitCount++;

}

}

file << "Кількість цифр: " << digitCount << "\n";

file << "Дата та число: " << dateTime;

file.close();

}

//=======================LAB10.3=======================

void binary(float x, float y, float z, unsigned int b, const string& outputFileName) {

ofstream outputsFile(outputFileName, ios\_base::app);

outputsFile << "Результат функцій ModulesHorbenko.h:" << endl;

outputsFile << "s\_calculation(" << x << ", " << y << ", " << z << "): " << s\_calculation(x, y, z) << endl;

outputsFile << "Число у двійковому коді: " << b << ": ";

for (int i = 31; i >= 0; i--) {

unsigned int mask = 1 << i;

outputsFile << ((b & mask) ? '1' : '0');

}

outputsFile << endl;

outputsFile.close();

}

***Лістинг TestDriver***

#include <cmath>

#include <iostream>

#include <fstream>

#include <string>

#include <conio.h>

#include "ModulesHorbenko.h"

using namespace std;

void testprocessfile(int test\_num, const string& inputFileName, const string& outputFileName)

{

cout << "\nTest\_" << test\_num << endl;

cout << "====================================" << endl;

processfile(inputFileName, outputFileName); // Використання переданих значень

cout << "Перевірити виконання того що тест-сьюта та результата програми, \nякщо воно сходиться натисніть 'y', якщо ні то іншу кнопку "<< endl;

char button = getch();

if (button == 'y')

{

cout << "Passed" << "\n====================================" << endl;

}else

{

cout << "Failed" << "\n====================================" << endl;

}

}

void testappendfileInfo(int test\_num, const string& filename)

{

cout << "\nTest\_" << test\_num << endl;

cout << "====================================" << endl;

appendfileInfo(filename);

cout << "Перевірте зміст файлу з тест-сьютом, \nякщо воно сходиться натисніть 'y', якщо ні то іншу кнопку " << endl;

char button = getch();

if (button == 'y')

{

cout << "Passed" << "\n====================================" << endl;

}else

{

cout << "Failed" << "\n====================================" << endl;

}

}

void testbinary(int test\_num, float x, float y, float z, unsigned int b, const string& outputFileName)

{

cout << "\nTest\_" << test\_num << endl;

cout << "====================================" << endl;

binary(x, y, z, b, outputFileName);

cout << "Перевірте зміст файлу з тест-сьютом, \nякщо воно сходиться натисніть 'y', якщо ні то іншу кнопку " << endl;

char button = getch();

if (button == 'y')

{

cout << "Passed" << "\n====================================" << endl;

}

else

{

cout << "Failed" << "\n====================================" << endl;

}

}

int main(){

system("chcp 1251 & cls");

cout << "=============Test\_10.1==============" << endl;

testprocessfile(1, "input\_1.txt", "output\_1.txt");

testprocessfile(2, "input\_2.txt", "output\_2.txt");

testprocessfile(3, "input\_3.txt", "output\_3.txt");

testprocessfile(4, "input\_4.txt", "output\_4.txt");

testprocessfile(5, "input\_5.txt", "output\_5.txt");

cout << "=============Test\_10.2==============" << endl;

testappendfileInfo(1, "input\_1.txt");

testappendfileInfo(2, "input\_2.txt");

testappendfileInfo(3, "input\_3.txt");

testappendfileInfo(4, "input\_4.txt");

testappendfileInfo(5, "input\_5.txt");

cout << "=============Test\_10.3==============" << endl;

testbinary(1, 5, 13, 1, 1, "binary\_1.txt");

testbinary(2, 5, 2, 0, 1567890, "binary\_2.txt");

testbinary(3, 10, 2, 5, 56, "binary\_3.txt");

testbinary(4, 50, 15, 100, 11, "binary\_4.txt");

testbinary(5, 3, 2, 5, 96, "binary\_5.txt");

return 0;

}