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”

На тему:

“ Реалізація програмних модулів оброблення даних складових типів з файловим введенням/виведенням”

ВИКОНАВ

студент академічної групи КІ-21-2

\_\_\_\_\_\_\_\_\_ А.М. Мельник

ПЕРЕВІРИВ

викладач кафедри кібербезпеки

та програмного забезпечення

\_\_\_\_\_\_\_\_\_ П.С. Усік

м. Кропивницький 2022

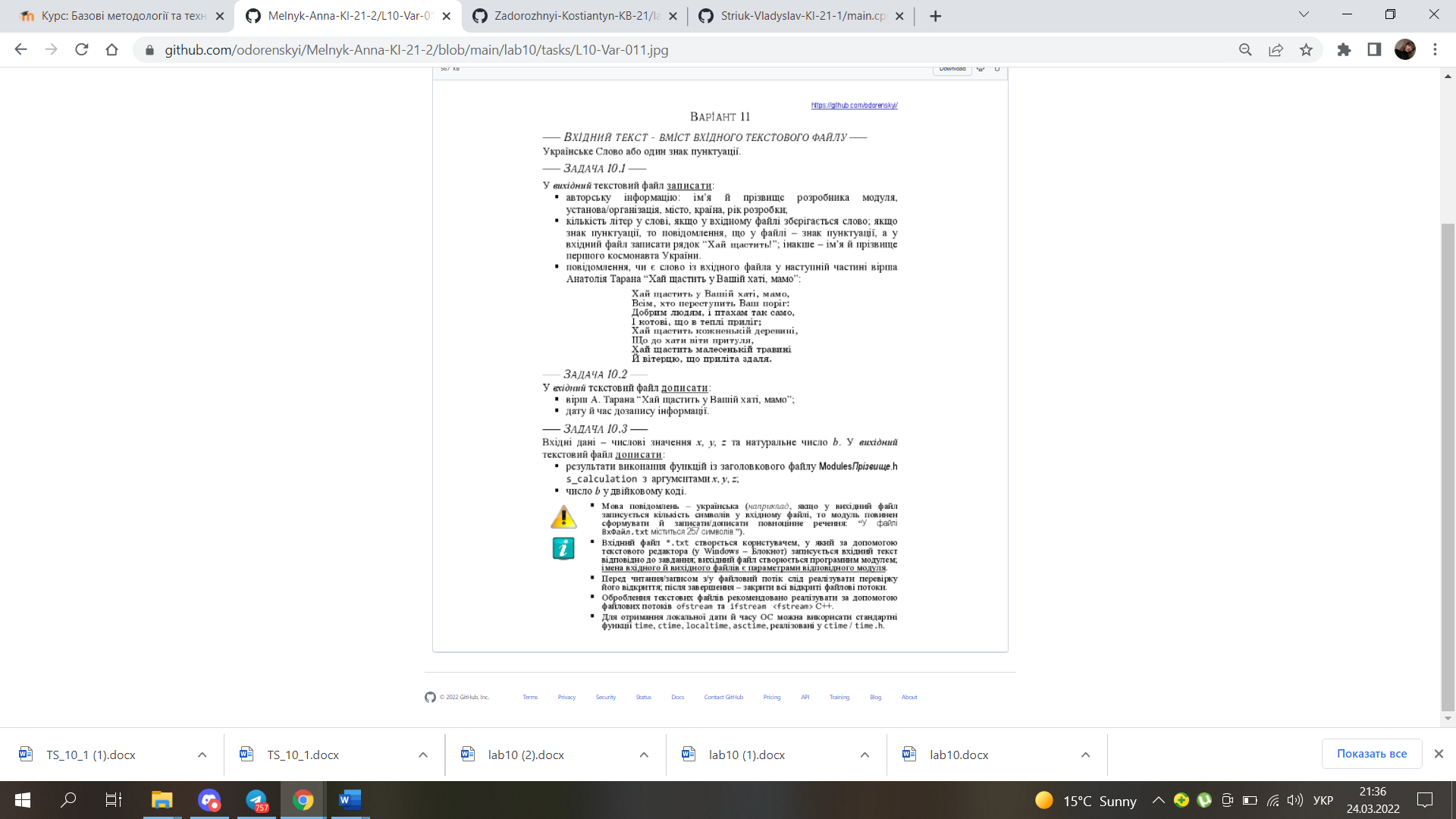
**Мета роботи :** полягає у набутті ґрунтовних вмінь і практичних навичок реалізації у Code::Blocks IDE мовою програмування С++ програмних модулів створення й оброблення даних типів масив, структура, об’єднання, множина, перелік, перетворення типів даних, використання файлових потоків та функцій стандартних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

**Завдання :**

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.

**Варіант 11**



**Задача 10.1**

У вхідний текстовий файл записуємо авторську інформацію, кількість літер у слові; якщо у вхідному файлі зберігається слово: якщо знак пунктуації, то повідомлення , що у файлі – знак пунктуації, а у в вхідний файл записати рядок “Хай щастить!”, інакше ім’я першого космонавта України; повідомлення чи є слово із вхідного файлу у наступні частині вірша Анатолія Тарана “Хай щастить у Вашій хаті, мамо”;

Хай щастить у Вашій хаті, мамо,

всім, хто переступіть Ваш поріг:

добрим людям, і птахам так само,

і котові, що в теплі приліг;

хай щастить кожненькій деревині,

що до хати віти притуля,

хай щастить малесенькій травині

й вітерцю, що приліта здаля.

**Аналіз задачі 10.1**

Записувати символи потрібіно у вхідний текстовий файл, результат отримуємо у вихідному файлі. Результат залежить від символів, які будуть розміщені у вхідному файлі.

**Постановка задачі 10.1**

Ми маємо два файли: вхідний так вихідний. У вхідний файл ми записуємо символи, або слово. Якщо це слово – тоді у вихідний текстовий файл зберігаємо кількість літер у слові, якщо знаки пунктуації – знак пунктуації, інакше – ім'я першого українського космонавта. Також здійснити пошук слова,яке ми зберегли у вихідному текстовому файлі у вірші Анатолія Тарана “Хай щастить у Вашій хаті, мамо”;

**Задача 10.2**

У вхідний текстовий файл дописати :

* вірш А. Тарана “Хай щастить у Вашій хаті, мамо”
* дату та час дозапису інформації

**Аналіз задачі 10.2**

Записувати інформацію потрібно лише у вхідний текстовий файл, помістити вірш , дату та час дозапису інформації.

**Постановка задачі 10.2**

Ми маємо лише вхідний текстовий файл, в який записуємо вірш, але для початку потрібно підключити препроцесорну директиву fstream та відкрити файл, далі ми розміщуємо вірш та при підключені препроцесорної директиви time.h зберігаємо дату та час дозапису інформації.

**Задача 10.3**

Вхідні дані – числові значення x та z та натуральне число b.У вихідний текстовий файл дозаписати :

* результати виконання заголовкової функції s\_calculation створеної в ModulesMelnik.h;
* число b в двійковій системі;

**Аналіз задачі 10.3**

Записувати інформацію потрібно у вихідний текстовий файл, дозаписати результати виконання заголовкової функції s\_calculation та число b в двійковій системі;

**Постановка задачі 10.3**

Для дозапису інформації в цьому завданні потрібен лише вихідний текстовий файл, для цього при виконанні тестового драйвера підключаємо бібліотеки для виконання заголовкової функції s\_calculation. Для дозапису

число b в двійковій системі створити функцію.

**Висновок:** з цієї лабораторної роботи я навчилася працювати з файлами. Для мене ця робота одна з найважчих, оскільки було багато нової інформації, яку потрібно використати.

Суть першому завдання була створити функції з інформацією про розробника, фунція з умовою до другого підпункту та пошук слова ,яке зчитується з файлу.Проблема виникла у потановці умови для другого пункту, оскільки, одна з умов не працювала, спочатку не виходило зчитати файл, якщо там букви, а потім через цикл при читанні слова, у файл виводилося безліч разів інформація про кількість літер.

У другому завданні проблем не виникло, оскільки там було просте завдання на дозапис вірша у файл, та написання інформації про дату та час дозапису файла.Для створення інформації про дату та час дозапису файла,потрібно підключити препроцесорну директиву та використати вже готову функцію \_\_TIMESTAMP\_\_ .

Суть третього завдання була у використанні функції, яку я створила у восьмій лабораторній роботі, та створенні функції для виконання перетворення числа з десятковї системи числення у двійкому. При створенні цієї функції хотілося знайти лаконічний дизайн, і я його знайшла. За допомогою декількох рядків програма перетворює число у двійкову систему числення. Щодо другого пункту- це я зробила першим з усіх завдань, але під час перевірки виник баг, в якому йшлося про те, що функції s\_calculation немає, тому було прийнято рішення у модулі створити нову функцію, з підключеннням старої та написання повної реалізації пункту у цій функції (у восьмій лабораторній роботі я створювала функцію, але там не було прописано те, що буде писатися в консольному вікні перед представленням вихідного числа).

Під час виконання тесту функцій виникла проблема, проект не хотів компілюватися, та після кожного натискання на початок компіляції створювався ще один файл, який перешкоджав компіляції, у проекті містилися приблизно 300 рядків про відомості препроцесорної директиви fstream, якщо закомітити цю директиву, то програма працювала, але якщо там не було роботи з файлами. Я намагалася у консольному вікні до гітхабу вписати команду git pull,яка видаляє все, до останнього файлу,який був завантажений на гітхаб.Для вирішення цієї проблеми я видалила програму та завантажила її знову, все запрацювало, скоріше за все, проблема була у лінкері.

Щодо завантаження роботи на git hub, в попередній роботі я робила це за допомогою консольного вікна, але проблема в тому, що можу редагувати декілька файлів одночасно, тому під час виконання 9 лабораторної роботи було прийнято рішення, що після виконання всієї роботи я вже готові папки завантажувала через сайт. Також виникла проблема при читанні коду: в MelnikTask та TestDriver замість кирилиці написано символи, які не стосуються роботи, це через те, що робота для зручності використовує кодування Windows1251, але для читання роботи на git hubкраще використовувати UTF-8.

Для цього спочатку потрібно головний файл проекту відрити у блокноті та зберегти з кодуванням UTF-8 (раніше було ANSI).Також в коді потрібно пікдключити “system (chcp 866 & cls)”, і в налаштуваннях editor налаштувати на кодування UTF-8.

**Додаток А:**

**Лістинг модуля :**

int word(string path\_infile, string path\_outfile){

string word = readfile(path\_infile);

ofstream in\_file(path\_infile, ios::app);

ofstream out\_file(path\_outfile, ios::app);

if(!in\_file.is\_open())

return -1;

if(!out\_file.is\_open())

return -1;

int b=0;

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

if (word[i] == ',' || word[i] == '.' || word[i] == '?' || word[i] == '!' ){

out\_file << "\nВ файлі присутня пунктуація";

in\_file << "\nХай щастить!";

}

if ((word[i]>='а' && word[i] <= 'я') || (word[i]>='А' && word[i] <= 'Я')) {

b++;

in\_file << "\n Кількість букв: \n" << b;

out\_file <<" Перший космонавт незалежної України - Леонід Каденюк";

}

}

in\_file.close();

out\_file.close();

return 0;

}

int findword(string path\_infile, string word)

{

ofstream in\_file(path\_infile, ios::app);

string poem = { "Хай щастить у Вашій хаті, мамо,\n"

"Всім, хто переступіть Ваш поріг:\n"

"Добрим людям, і птахам так само,\n"

"І котові, що в теплі приліг;\n"

"Хай щастить кожненькій деревині,\n"

"Що до хати віти притуля,\n"

"Хай щастить малесенькій травині\n"

"Й вітерцю, що приліта здаля.\n" };

size\_t fw;

fw = poem.find(word);

if(!in\_file.is\_open())

return -1;

if (fw != string::npos) {

in\_file << "Слово \"" << word << "знайдено" << endl << endl;

in\_file.close();

return true;

}

in\_file << "Cлово " << word << " - не знайдено" << endl << endl;

in\_file.close();

return false;

}

string readfile(string path\_file){

ifstream file(path\_file);

if(!file.is\_open()) return "ERROR";

string text = "";

string line;

while (getline(file, line)){

text += line + '\n';

}

file.close();

return text;

}

int poem(string pathfile){

ofstream file(pathfile,ios::app);

if(!file.is\_open())

return -1;

file << "Ви завжди за всіх були дбайливі -"<< endl

<< "Бо така ж та доля у вдови."<< endl

<< "Бо ж як будуть всі навкруг щасливі,"<< endl

<< "То щасливі будете і Ви."<< endl

<< "Дата та час дозапису інформації:" << endl << endl

<< \_\_TIMESTAMP\_\_<< endl;

file.close();

return 0;

}

void task10\_1(string path\_infile, string path\_outfile){

info(path\_infile);

word(path\_infile,path\_outfile);

// findword(path\_infile,word)

}

void task10\_2(string pathfile){

poem(pathfile);

}

void task10\_3(string pathfile){

calc(pathfile);

binary(pathfile, 1337);

}

**Лістинг тестдрайвера:**

#include <iostream>

#include <fstream>

#include <ctime>

#include <cstring>

#include <math.h>

#include <tgmath.h>

#include <windows.h>

#include <string>

#include <bitset>

#include "ModulesMelnik.h"

using namespace std;

string readfile(string path\_file);

void testdriver\_1(void);

void testdriver\_2(void);

void testdriver\_3(void);

string clearText(string out\_path);

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

testdriver\_1();

testdriver\_2();

testdriver\_3();

system("pause");

return 0;

}

void testdriver\_1(void){

char in\_path[] = "D:\\.git\\Melnyk-Anna-KI-21-2\\lab10\\TestFile\\228.txt";

char out\_path[] = "D:\\.git\\Melnyk-Anna-KI-21-2\\lab10\\\TestFile\\1337.txt";

string expected\_test\_10\_1\_1 = "D:\\.git\\Melnyk-Anna-KI-21-2\\lab10\\TestFile\\in\_file1.txt";

string expected\_test\_10\_1\_2 = "D:\\.git\\Melnyk-Anna-KI-21-2\\lab10\\TestFile\\in\_file.txt";

string reader, result\_task\_10\_1\_1, Rexpected\_test\_10\_1\_1;

task10\_1(in\_path, out\_path);

Rexpected\_test\_10\_1\_1 = readfile(expected\_test\_10\_1\_1);

reader = readfile(out\_path);

cout << " Test10\_1 " << endl;

if(reader == Rexpected\_test\_10\_1\_1) cout << "\nPASSED.\n";

else cout << "\n FAILED.\n";

}

void testdriver\_2(void){

time\_t rawtime;

time(&rawtime);

char in\_path[] = "D:\\.git\\Melnyk-Anna-KI-21-2\\lab10\\prj\\228.txt";

char out\_path[] = "D:\\.git\\Melnyk-Anna-KI-21-2\\lab10\\prj\\1337.txt";

string expected\_test\_10\_2\_1 = "D:\\.git\\Melnyk-Anna-KI-21-2\\lab10\\TestFile\\in\_file2.txt";

string reader, result\_task\_10\_2\_1, Rexpected\_test\_10\_2\_1;

task10\_2(in\_path);

Rexpected\_test\_10\_2\_1 = readfile(expected\_test\_10\_2\_1);

Rexpected\_test\_10\_2\_1 += ctime(&rawtime);

reader = readfile(in\_path);

cout << "\n Test10\_2 " << endl;

if(reader == Rexpected\_test\_10\_2\_1) cout << "\nPASSED.\n";

else cout << "\nFAILED.\n";

}

void testdriver\_3(void){

char in\_path[] = "D:\\.git\\Melnyk-Anna-KI-21-2\\lab10\\prj\\228.txt";

char out\_path[] = "D:\\.git\\Melnyk-Anna-KI-21-2\\lab10\\prj\\1337.txt";

string expected\_path\_10\_3\_1 = "D:\\.git\\Melnyk-Anna-KI-21-2\\lab10\\TestFile\\in\_file3.txt";

string expected\_path\_10\_3\_2 = "D:\\.git\\Melnyk-Anna-KI-21-2\\lab10\\TestFile\\in\_file3\_2.txt";

string reader;

float x[5] = { 1, 9, 7, 8, 10};

float z[5] = { 8, 5, 1, 2, 5};

int num [5] = {27, 35,4,328,367};

double expectedResult[5] = {76.7261, 101.871, 91.012, 44.1321, 92.1105};

cout << " Test10\_3 " << endl;

for (short i = 0; i < 5; i++)

{

if (round(s\_calculation(x[i], z[i])\*10000)/10000.0 == expectedResult [i])

{

cout << i + 1 << " PASSED\n";

}

else {

cout << i + 1 << " Failed\n";

}

reader = readfile(out\_path);

}

}

|  |  |  |
| --- | --- | --- |
| input text | Expect result | Result |
| ааапподівдо  Кількість букв 11  Слово не знайдено | ааапподівдо  Кількість букв 11  Слово не знайдено | passed |
| мамо  Кількість букв 4  Слово знайдено | мамо  Кількість букв 4  Слово знайдено | passed |
| щастить  Кількість букв 7  Слово знайдено | щастить  Кількість букв 7  Слово знайдено | passed |
| віпвд  Кількість букв 5  Слово не знайдено | віпвд  Кількість букв 5  Слово не знайдено | passed |
| хай  Кількість букв 3  Слово знайдено | хай  Кількість букв 3  Слово знайдено | passed |

**Додаток B**

**TS-1**

**TS-2**

|  |  |  |
| --- | --- | --- |
| input text | Action (test steps) | Expected Result |
| Ви завжди за всіх були дбайливі –  Бо така ж та доля у вдови.  Бо ж як будуть всі навкруг щасливі,  то щасливі будете і Ви.  Дата та час дозапису інформації:  Tue Mar 29 19:48:14 2022 | Ви завжди за всіх були дбайливі –  Бо така ж та доля у вдови.  Бо ж як будуть всі навкруг щасливі,  то щасливі будете і Ви.  Дата та час дозапису інформації:  Tue Mar 29 19:48:14 2022 | passed |

**TS-3**

|  |  |  |
| --- | --- | --- |
| Preliminary Steps | Action (test steps) | Expected Result |
| Результат функції:8.50368  Число в двійковій системі:  110 | Введіть x:  5  Введіть z:  4  Введіть num:  6 | passed |
| Результат функції:65.5273  Число в двійковій системі:  110000101 | Введіть x:  7  Введіть z:  2  Введіть num:  389 | passed |
| Результат функції:20093.2  Число в двійковій системі:  1111011 | Введіть x:  3  Введіть z:  23  Введіть num:  123 | passed |
| Результат функції:13013.9  Число в двійковій системі:  1010000111 | Введіть x:  9  Введіть z:  56  Введіть num:  647 | passed |
| Результат функції:553417  Число в двійковій системі:  1010000111 | Введіть x:  45  Введіть z:  678  Введіть num:  999 | passed |