

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

КБ-23 Чернолєс К.С.

ПЕРЕВІРИВ

викладач кафедри кібербезпеки

та програмного забезпечення

Дресєва Г.М.

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 C++ 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 символної інформації.

Варіант 2

Завдання:

1. Реалізувати програмні модулі розв'язування задач 10.1–10.3 як складові статичної бібліотеки libModulesПрізвище.a (проект ModulesПрізвище лабораторних робіт №8–9).
2. Реалізувати тестовий драйвер автоматизованої перевірки програмних модулів розв'язування задач 10.1–10.3.

ХІД РОБОТИ

Завдання 10.1

— ЗАДАЧА 10.1 —

У вихідний текстовий файл записати:

- авторську інформацію: ім'я й прізвище розробника модуля, установа/організація, місто, країна, рік розробки;
- кількість голосних літер у слові з вхідного файлу;
- повідомлення, чи є слово із вхідного файлу в наступній крапинці Віталія Іващенко:

До щастя не пускає лінощів орава.
У чім воно — ніхто не знає до пуття.
Навчитись радісно робити кожну справу —
Найперше правило *щасливого життя*.

Рисунок 1.1 - Умова задачі 10.1

Строга постановка задачі:

Вхідні дані: слово в форматі string записане до вхідного файлу

Вихідні дані: записані до вихідного файлу: авторська інформація, кількість голосних літер (натуральне число), результат пошуку слова в поемі (bool).

Проектування архітектури програмного модуля:

authorInfo() - приймає в якості аргументу назву вихідного файлу. Виконується очищення вмісту файлу та запис авторської інформації.

vowelsCountInFile() - отримує на вхід назву вихідного файлу та слово, що міститься у вхідному файлі. Цикл в 10 ітерацій реалізовано для перевірки наявності голосних літер, що ініціалізовані як рядковий масив, кожен елемент якого - голосна літера українського алфавіту. На вихід отримуємо дозапис у вихідний файл кількості голосних літер.

findWordInPoem() - отримує на вхід назву вихідного файлу та слово, що міститься у вхідному файлі. Поема ініціалізована як рядок, котрий надалі використовується в функції бібліотеки <cstring> для пошуку вхідного слова. Істинність пошуку визначається співпадінням позиції

каретки у файлі с позицією результату стандартної функції `find()`. До вихідного файлу записується результат пошуку.

Завдання 10.2

— ЗАДАЧА 10.2 —

У вхідний текстовий файл дописати:

- кількість приголосних літер у ньому, дату й час дозапису інформації.

Рисунок 1.2 - Умова задачі 10.2

Строго постановка задачі:

Вхідні дані: слово в форматі `string` записане до вхідного файлу

Вихідні дані: записані до вихідного файлу: кількість приголосних літер (натуральне число), дата та час дозапису.

Проектування архітектури програмного модуля:

consonantsCountInFile() - приймає на вхід назву вхідного файлу та слово, що міститься в ньому. Спираючись на реалізацію функції пошуку голосних літер, повторюємо цикл з ***vowelsCountInFile()***, проте після кожної ітерації видаляємо з слова-аргументу знайдену голосну літеру. Далі, кількість приголосних визначається довжиною слова-аргумента без голосних літер. У вхідний файл ведеться дозапис кількості приголосних в ньому.

timestampInFile() - приймає аргумент назву вхідного файлу. В даний файл записується дата та час дозапису інформації за допомогою стандартної функції з бібліотеки `<ctime>`.

Завдання 10.3

— ЗАДАЧА 10.3 —

Вхідні дані – числові значення *x*, *y*, *z* та натуральне число *b*. У вихідний текстовий файл дописати:

- результати виконання функцій із заголовкового файлу `Modules/Прізвище.hs_calculation` з аргументами *x*, *y*, *z*;
- число *b* у двійковому коді.

Рисунок 1.3 - Умова задачі 10.3

Строга постановка задачі:

Вихідні та вхідні дані вказані в умові задачі.

Проектування архітектури програмного модуля:

sCalculationResInFile() - аргументи: назва вихідного файлу та три цілочисельних числа. До вихідного файлу записується результат виконання функції ***s_caculation***, що приймає в якості аргументів три цілочисельних значення.

decimalToBinary() - вхідні дані: назва вихідного файлу та натуральне число. Оголошується масив розміром в 32 елементи (стільки бітів вміщує змінна типу ***int***) і в циклі, з використанням оператора ***'%'*** заповнюємо масив остачами від ділення на 2. В наступному циклі виводимо поелементно, починаючи з останнього, вміст масиву до вихідного файлу.

Результат виконання **TestDriver.exe**:

Запис слова до вхідного файлу:	true
10.1(1) - Запис авторської інформації до вихідного файлу:	true
10.1(2) - Результат підрахунку кількості голосних літер та дозапис до вихідного файлу:	true
10.1(3) - Результат пошуку слова та дозапис до вихідного файлу:	true
10.2(1) - Результат підрахунку кількості приголосних літер та дозапис до вхідного файлу:	true
10.2(2) - Дозапис часу редагування файлу до вхідного файлу:	true
10.3(1) - Дозапис результату функції <i>s_caculation</i> до вихідного файлу:	true
10.3(1) - Результат конвертування десяткового числа в двійкове та дозапис до вихідного файлу:	false
Запис слова до вхідного файлу:	true
10.1(1) - Запис авторської інформації до вихідного файлу:	true
10.1(2) - Результат підрахунку кількості голосних літер та дозапис до вихідного файлу:	true
10.1(3) - Результат пошуку слова та дозапис до вихідного файлу:	false
10.2(1) - Результат підрахунку кількості приголосних літер та дозапис до вхідного файлу:	true
10.2(2) - Дозапис часу редагування файлу до вхідного файлу:	true
10.3(1) - Дозапис результату функції <i>s_caculation</i> до вихідного файлу:	true
10.3(1) - Результат конвертування десяткового числа в двійкове та дозапис до вихідного файлу:	true
Запис слова до вхідного файлу:	true
10.1(1) - Запис авторської інформації до вихідного файлу:	true
10.1(2) - Результат підрахунку кількості голосних літер та дозапис до вихідного файлу:	true
10.1(3) - Результат пошуку слова та дозапис до вихідного файлу:	true
10.2(1) - Результат підрахунку кількості приголосних літер та дозапис до вхідного файлу:	true
10.2(2) - Дозапис часу редагування файлу до вхідного файлу:	true
10.3(1) - Дозапис результату функції <i>s_caculation</i> до вихідного файлу:	true
10.3(1) - Результат конвертування десяткового числа в двійкове та дозапис до вихідного файлу:	true
Запис слова до вхідного файлу:	true
10.1(1) - Запис авторської інформації до вихідного файлу:	true
10.1(2) - Результат підрахунку кількості голосних літер та дозапис до вихідного файлу:	true
10.1(3) - Результат пошуку слова та дозапис до вихідного файлу:	false
10.2(1) - Результат підрахунку кількості приголосних літер та дозапис до вхідного файлу:	true
10.2(2) - Дозапис часу редагування файлу до вхідного файлу:	true
10.3(1) - Дозапис результату функції <i>s_caculation</i> до вихідного файлу:	true
10.3(1) - Результат конвертування десяткового числа в двійкове та дозапис до вихідного файлу:	true

Запис слова до вхідного файлу:	true
10.1(1) - Запис авторської інформації до вихідного файлу:	true
10.1(2) - Результат підрахунку кількості голосних літер та дозапис до вихідного файлу:	true
10.1(3) - Результат пошуку слова та дозапис до вихідного файлу:	true
10.2(1) - Результат підрахунку кількості приголосних літер та дозапис до вхідного файлу:	true
10.2(2) - Дозапис часу редагування файлу до вхідного файлу:	true
10.3(1) - Дозапис результату функції <code>s_calculation</code> до вихідного файлу:	true
10.3(1) - Результат конвертування десяткового числа в двійкове та дозапис до вихідного файлу:	true

ВИСНОВКИ

Проаналізувавши завдання лабораторної роботи та користуючись поняттям модульності програмного забезпечення було прийнято рішення про градування кожного завдання на підфункції, що сконцентровані на вирішенні конкретної, логічної частини завдання. Метою розбиття функцій є досягнення більш детального декларування результатів тестування окремих частин програми.

Для виведення результатів виконання функцій зі статичної бібліотеки `labModulesChernoles.a` у консольне вікно застосунку `TestDriver.exe` було використано маніпулятор потокового виводу `boolalpha` із стандартної бібліотеки `<iomanip>`. Таким чином, підтверджується рівність очікуваних даних з отриманими у випадку, якщо функція повертає значення не типу `bool`. Функції, з результатом повернення `bool` визначають, чи є, насамперед, істинним виконання даної функції, чи ні.

Враховано коректне відкриття та закриття файлових потоків для всіх функцій. Так, на початку функціонування кожної процедури оголошується змінна типу, відповідного до мети взаємодії файлових потоків (`ofstream` - для запису / `ifstream` - для читання). Додатково, в аргументах даних змінних, окрім вказання назви файлу, вказано тип відкриття файлу (`ios::out` - для запису / `ios::in` - для зчитування / `ios::app` - для дозапису). В режимі `ios::out` працюють лише ті процедури, що мають виконати перезапис всього файлу (тобто функції, що вперше взаємодіють з файлом).

Аби запобігти повторенню однакової частини коду в кожній функції, на початку функціонування `TestDriver.exe` викликається

`abilityToEdit()`, що перевіряє обидва файли на можливість до відкриття / редагування.

Також, задля автоматизації процесу була опущена умова лабораторної роботи щодо необхідності запису будь-якого українського слова до вхідного файлу власноруч. Натомість реалізовано додаткову процедуру, що за прийнятим аргументом в якості будь-якого українського слова виконує його запис до вхідного файлу.

Оперуючи вимогою щодо локалізації програмного забезпечення українською мовою, була виправлена проблема з ініціалізацією змінних типу `char` що мали містити в собі символи кирилиці. Так як це неможливо зробити, через особливості типу, запис кириличних символів, надалі, відбувається у змінних типу `string`.

Пошук голосних, приголосних літер та слова у поемі здійснюється завдяки використанню змінної `size_t` в котру записувався результат пошуку підрядка в рядку. Надалі виконується порівняння з константним значенням `string::npos` (вміст не має дорівнювати цій константі). Якщо умова виконується - пошук вважається успішним.

Використовуючи реалізовану функцію підрахунку голосних літер, було створено процедуру підрахунку приголосних літер. Її роботу дія базується на тому, аби за шаблоном визначити положення голосних літер та за допомогою стандартної функції `erase()` з бібліотеки `<cstring>` - видалити їх. Так, отримавши рядок без голосних, можна бути певним, що всі його символи - приголосні літери. Тому для виведення фінального результату, достатньо використати функцію `length()` (довжина рядка).

Функція визначення двійкового представлення числа основана на стандартному ("ручному") методі переведення числа з десяткової системи числення до двійкової. Врахувавши, що число типу `int` вміщує в собі 32 біти, оголошуємо масив з відповідним розміром для його подальшого заповнення двійковими 0 та 1. В першому циклі вхідне число поступово ділиться на 2 з отриманням остачі (використовується оператор `%`) та ділиться на 2 звичайним методом для подальшої ітерації. Цикл

завершується, коли числа стає рівне 0. Так як біти в числі рахуються зправа-наліво, створено наступний цикл, в котрому відбувається поелементний запис до вихідного файлу елементів масиву, починаючи з останнього і закінчуючи початковим.

ДОДАТОК А

(Вихідний код ModulesChernoles)

... / ModulesChernoles / main.cpp

```
#include <iostream>
#include <fstream>
#include <bitset>
#include <ctime>
#include <sstream>
#include <cstring>

int vowelsCountInFile string outputFileName string ukrWord

    outputFile ios
    vowels 10 = "a" "e" "e" "и" "i" "i" "o" "y" "ю" "я"
int
    size_t = 0

    for int = 0 < 10 ++
        = ukrWord find vowels
        if != string
            ++

        << "Кількість голосних літер з вхідного файлу: " << <<

    outputFile close
    return

bool findWordInPoem string outputFileName string ukrWord

    outputFile ios
    = "до щастя не пускає лінощів орава.\n"
    "у чім воно - ніхто не знає до пуття.\n"
    "навчитись радісно робити кожну справу\n"
    "найперше правило щасливого життя\n"

    size_t

    = poemVI find
    if != string
        << "Знайдено слово \"" << << "\"" << <<
        outputFile close
        return true

        << "Слово \"" << << "\" - не знайдено" << <<
    outputFile close
    return false

int consonantsCountInFile string inputFileName string ukrWord

    inputTxtFile ios
    vowels 10 = "a" "e" "e" "и" "i" "i" "o" "y" "ю" "я"
    size_t

    for int = 0 < 10 ++
        = ukrWord find vowels
        if != string
            ukrWord erase vowels length

        << "Кількість приголосних літер з вхідного файлу: " <<
    ukrWord length / 2 << <<
    inputTxtFile close
```

```

        return ukrWord length / 2

bool timestampInFile string inputFileName

        inputFile ios
        time_t
        time &
        << "Дата та час дозапису інформації: " << ctime & << <<

        inputFile close
        return true

double sCalculationResInFile string outputFileName float x float y float z

        outputFile "prjOutputFile.txt" ios
        << "Результат виконання s_calculation: " << s_calculation <<
<<
        outputFile close
        return s_calculation

bool decimalToBinary string outputFileName unsigned int number

        if <= 0
            return false

        outputFile ios
        int binaryCels 32
        int
        unsigned int =

        for = 0 > 0 ++
            binaryCels = % 2
            = / 2

        << "Двійкове представлення числа " << << " : "
        for = - 1 >= 0 --
            << binaryCels

        << <<
        outputFile close
        return true

bool authorInfo string outputFileName

        outputFile ios
        <<
        "===== " <<
        << " Виконавець: Чернолес Кирило Сергійович" <<
        << " Рік розробки: 2023" <<
        << " Місто/Країна: Кропивницький/Україна" <<
        << " ВНЗ: Центральний Національний Технічний Університет"
<<
        <<
        "===== " << <<
        outputFile close
        return true

```

... / ModulesChernoles / ModulesChernoles.h

```
#ifndef MODULESCHERNOLES_H_INCLUDED
#define MODULESCHERNOLES_H_INCLUDED

using namespace std

float s_calculation float x float y float z

bool authorInfo string outputFileName
int vowelsCountInFile string outputFileName string ukrWord
bool findWordInPoem string outputFileName string ukrWord
int consonantsCountInFile string inputFileName string ukrWord
bool timestampInFile string inputFileName
double sCalculationResInFile string outputFileName float x float y float z
bool decimalToBinary string outputFileName unsigned int number

#endif // MODULESCHERNOLES_H_INCLUDED
```

(Вихідний код TestDriver)

... / TestDriver / main.cpp

```
                << sCalculationResInFile "prjOutputFile.txt" sCalcX sCalcY
sCalcZ    <= expectedResult    + 0.005
                <<
                <<
                << "10.3(1) - Результат конвертування десяткового числа в двійкове та
дозапис до вихідного файлу: \t"
                << decimalToBinary "prjOutputFile.txt" naturalDigit
                <<      <<      <<

        _getch

return 0
```

ДОДАТОК В
(TestSuite до завдання 10.1)

Назва тестового набору / Test Suite Description	TS_MODULE
Назва проекту/ПЗ / Name of project	TestDriver.exe
Рівень тестування / Level of testing	Модульний
Автор тест-сьюта / Test Suite Author	Чернолес Кирило
Виконавець / Implementer	Чернолес Кирило

Test Case ID	Actions	Expected Result	Test Result
TS_1	1. Запис "щастя" у prjInputFile.txt 2. Запис інформації у prjOutputFile.txt 3. Запис кількості голосних у prjOutputFile.txt 4. Запис пошуку "щастя" у prjOutputFile.txt	Вміст prjInputFile.txt: щастя Вміст prjOutputFile.txt: =====	Passed
TS_2	1. Запис "воля" у prjInputFile.txt 2. Запис інформації у prjOutputFile.txt 3. Запис кількості голосних у prjOutputFile.txt 4. Запис пошуку "воля" у prjOutputFile.txt	Вміст prjInputFile.txt: воля Вміст prjOutputFile.txt: =====	Passed
TS_3	1. Запис "правило" у prjInputFile.txt 2. Запис інформації у prjOutputFile.txt 3. Запис кількості голосних у prjOutputFile.txt 4. Запис пошуку "правило" у prjOutputFile.txt	Вміст prjInputFile.txt: правило Вміст prjOutputFile.txt: =====	Passed

		<div>Рік розробки: 2024</div> <div>Місто/Країна: Кропивницький/Україна</div> <div>ВНЗ: Центральний Національний Технічний Університет</div> <div>=====</div> <div>Кількість голосних літер з вхідного файлу: 3</div> <div>Знайдено слово " правило"</div>	
TS_4	<div>1. Запис " праця" у prjInputFile.txt</div> <div>2. Запис інформації у prjOutputFile.txt</div> <div>3. Запис кількості голосних у prjOutputFile.txt</div> <div>4. Запис пошуку " праця" у prjOutputFile.txt</div>	<div>Вміст prjInputFile.txt:</div> <div>праця</div> <div>Вміст prjOutputFile.txt:</div> <div>=====</div> <div>Виконавець: Чернолес Кирило</div> <div>Сергійович</div> <div>Рік розробки: 2024</div> <div>Місто/Країна: Кропивницький/Україна</div> <div>ВНЗ: Центральний Національний Технічний Університет</div> <div>=====</div> <div>Кількість голосних літер з вхідного файлу: 2</div> <div>Знайдено слово " праця"</div>	Passed
TS_5	<div>1. Запис " життя" у prjInputFile.txt</div> <div>2. Запис інформації у prjOutputFile.txt</div> <div>3. Запис кількості голосних у prjOutputFile.txt</div> <div>4. Запис пошуку " життя" у prjOutputFile.txt</div>	<div>Вміст prjInputFile.txt:</div> <div>життя</div> <div>Вміст prjOutputFile.txt:</div> <div>=====</div> <div>Виконавець: Чернолес Кирило</div> <div>Сергійович</div> <div>Рік розробки: 2024</div> <div>Місто/Країна: Кропивницький/Україна</div> <div>ВНЗ: Центральний Національний Технічний Університет</div> <div>=====</div> <div>Кількість голосних літер з вхідного файлу: 2</div> <div>Знайдено слово " життя"</div>	Passed

ДОДАТОК Г

(TestSuite до завдання 10.2)

Назва тестового набору / Test Suite Description	TS_MODULE
Назва проекту/ПЗ / Name of project	TestDriver.exe
Рівень тестування / Level of testing	Модульний
Автор тест-сьюта / Test Suite Author	Чернолєс Кирило
Виконавець / Implementer	Чернолєс Кирило

Test Case ID	Actions	Expected Result	Test Result
TS_1	1. Запис кількості приголосних у prjInputFile.txt 2. Запис часу дозапису у prjInputFile.txt	Вміст prjInputFile.txt: щастя Кількість приголосних літер з вхідного файлу: 3 Дата та час дозапису інформації: [date]	Passed
TS_2	1. Запис кількості приголосних у prjInputFile.txt 2. Запис часу дозапису у prjInputFile.txt	Вміст prjInputFile.txt: воля Кількість приголосних літер з вхідного файлу: 2 Дата та час дозапису інформації: [date]	Passed
TS_3	1. Запис кількості приголосних у prjInputFile.txt 2. Запис часу дозапису у prjInputFile.txt	Вміст prjInputFile.txt: правило Кількість приголосних літер з вхідного файлу: 4 Дата та час дозапису інформації: [date]	Passed
TS_4	1. Запис кількості приголосних у prjInputFile.txt 2. Запис часу дозапису у prjInputFile.txt	Вміст prjInputFile.txt: праця Кількість приголосних літер з вхідного файлу: 3 Дата та час дозапису інформації: [date]	Passed
TS_5	1. Запис кількості приголосних у prjInputFile.txt 2. Запис часу дозапису у prjInputFile.txt	Вміст prjInputFile.txt: життя Кількість приголосних літер з вхідного файлу: 3 Дата та час дозапису інформації: [date]	Passed

ДОДАТОК Д
(TestSuite до завдання 10.3)

Назва тестового набору / Test Suite Description	TS_MODULE
Назва проекту/ПЗ / Name of project	TestDriver.exe
Рівень тестування / Level of testing	Модульний
Автор тест-сьюта / Test Suite Author	Чернолєс Кирило
Виконавець / Implementer	Чернолєс Кирило

Test Case ID	Actions	Expected Result	Test Result
TS_1	1. Запис результату s_calculation (6.9, 3, -0.8) у prjOutputFile.txt 2. Запис двійкового числа 0 у prjOutputFile.txt	Вміст prjOutputFile.txt: [...] Результат виконання s_calculation: 6.01205	Passed
TS_2	1. Запис результату s_calculation (3.65, -5, 0.8) у prjOutputFile.txt 2. Запис двійкового числа 23 у prjOutputFile.txt	Вміст prjOutputFile.txt: [...] Результат виконання s_calculation: 5.54619 Двійкове представлення числа 23 : 10111	Passed
TS_3	1. Запис результату s_calculation (10, 0.3, 0) у prjOutputFile.txt 2. Запис двійкового числа 42 у prjOutputFile.txt	Вміст prjOutputFile.txt: [...] Результат виконання s_calculation: 7.87711 Двійкове представлення числа 42 : 101010	Passed
TS_4	1. Запис результату s_calculation (0.2, -0.3, 4) у prjOutputFile.txt 2. Запис двійкового числа 100 у prjOutputFile.txt	Вміст prjOutputFile.txt: [...] Результат виконання s_calculation: 4.08708 Двійкове представлення числа 100 : 1100100	Passed
TS_5	1. Запис результату s_calculation (39, 9, -4) у prjOutputFile.txt 2. Запис двійкового числа 699 у prjOutputFile.txt	Вміст prjOutputFile.txt: [...] Результат виконання s_calculation: 14.469 Двійкове представлення числа 699: 1010111011	Passed